Cosa contiene la Cassetta?

C = 16:
1. Poldo
2. Diablo
3. Golf
4. Poker
5. Crazy Train
6. Battleship

MSX:
1. Digger
2. Poyan
3. Camelot
4. Runner
5. Magical
6. Aliens
7. L'isola

sommario

C16/MSX E' IN EDICOLA OGNI VENERDI' DELL'ULTIMA SETTIMANA DEL MESE

Desidero abbonarmi alla rivista 16/MSX allo speciale prezzo di £ 100.000 per 10 copie.

COGNOME ______________ NOME ______________

VIA ______________

CAP ______ CITTÀ ______________ PROV. ______

Allego assegno ☐ ric. versamento sul c/c postale n. 11319209 ☐ intestato a Gruppo Editoriale International Education.

Le richieste di abbonamento andranno inoltrate a:
Gruppo Editoriale International Education
Viale Famagosta, 75 - 20142 Milano - tel. 89502256

Da oggi potrai abbonarti a 16/MSX e ricevere la tua rivista comodamente a casa semplicemente sottoscrivendo uno speciale abbonamento per 10 numeri allo specialissimo prezzo di £ 100.000 anziché £ 110.000.
Potrai così assicurarti la tua copia, risparmiare subito 10.000 lire e avere la sicurezza del prezzo bloccato per tutta la durata dell'abbonamento.

GAMES VIDEO GAMES

AVVERTENZE

Questa cassetta è stata registrata con cura e con i più alti standard di qualità.
Leggete con attenzione le istruzioni per il caricamento. Nel caso in cui, per una ragione qualsiasi, trovaste difficoltà nel caricare i programmi, spedite la cassetta al seguente indirizzo:

Gruppo Editoriale International Education srl
Viale Famagosta, 75
20142 Milano

Testeremo il prodotto e, nel caso, lo sostituiremo con uno nuovo senza costi supplementari.

GAMES VIDEO GAMES

attenzione! attention! look out! achtung!

Occhio all'azimut

Per la buona lettura della cassetta occorre che la testina del registratore sia pulita ed allineata col nastro. Se così non fosse potrebbe accadere che sul video appaia "error". Pulite allora la testina del registratore con un cottonfioc imbevuto di alcool. Se nonostante questa operazione il computer continua a non caricare bene prendete un cacciavite ed agite direttamente (in senso orario o antiorario) sulla vite apposita di regolazione del-l'azimut.

Se avete un Commodore 16 *digitate LOAD e RETURN, quindi avviate il registratore.*
Per un buon caricamento dei programmi è opportuno tenere il registratore lontano dal monitor e dall'alimentatore.

Se siete i possessori di un MSX *per caricare ogni singolo programma digitate RUN "CAS:" e RETURN.*
Ogni eventuale variazione apparirà in calce alle recensioni dei giochi alla rubrica MSX Challenge.

Recensione libri

MSX LINGUAGGIO E PROGRAMMAZIONE
di Ekkehard KAIER
Edito da Tecniche Nuove
Pagine: 372
Prezzo: 43.000 lire

Questa volta vi proponiamo un libro dalle dimensioni decisamente inconsuete per un libro sull'MSX e dal contenuto interessantissimo.
Il libro è diviso in più parti indipendenti tra loro. Ovviamente, la prima parte è di introduzione all'informatica e può essere saltata a pie' pari da chi è già esperto.
Questa prima parte è davvero ottima e introduce molto bene i diversi concetti fondamentali, spiegando cosa sono hardware, software e firmware, cosa sono i grossi calcolatori e i microcomputer, le strutture di dati e le strutture di programma, i sistemi operativi e i programmi applicativi, come si installa un programma già pronto e come implica programmare. Dopo la lettura di questa parte si è messi in grado di inquadrare i computer MSX nel panorama informatico.
Il manuale continua poi con l'MSX spiegando come si usano tastiera, monitor, floppu, drive e stampanti MSX, come si scribe il primo programma e quali comandi contiene l'MSX BASIC, qual'è la differenza tra lavoro a livello di MSX-BASIC e al livello del sistema operativo MSX-DOS. Spiega poi i comandi dell'MSX-DOS con chiari esempi come viene fatto con la spiegazione dei comandi MSX-BASIC. Dopo la lettura di questa seconda sezione si è in grado di usare davvero il computer MSX laciando programmi e scrivendone di semplici in BASIC con la possibilità di memorizzarli.
Dopo queste due eccellenti sezioni, il manuale continua con un vero e proprio corso di programmazione che riguarda programmi che illustrano le varie strutture (sequenziali, condizionate, iterative, sottoprogrammi); l'elaborazione di testi e l'input/output; la programmazione in codice macchina e la ricerca, ordinamento e classificazione di dati; i file sequenziali e quelli ad accesso diretto; la grafica, gli sprite e gli effetti sonori.
Il libro è decisamente eccellente e contiene 101 programmi esemplificativi riportati e commentati accuratamente. I capitoli 2 e 3 si completano logicamente e devono essere letti in ordine di successione, mentre la lettura del capitolo 1 può procedere parallelamente.
In conclusione, questo libro è da non perdere: è adattassimo ai principianti, mentre per gli utenti più esperti è un riferimento indispensabile grazie soprattutto agli indici analitici e alle accurate tabelle riassuntive.

EDITORIALE

Quaranta numeri sono davvero tanti e se ci guardiamo alle possiamo dire che siamo cresciuti molto, soprattutto negli ultimi numeri grazie al vostro aiuto, al vostro interesse.
In fondo siamo rimasti solo noi a tenere vivo questo sistema, potente sistema. Una vera e propria Cenerentola informatica.
Ce l'ho con i produttori di MSX che hanno adottato la tecnica "Amstrad CPC" del vendi e fuggi, ma ce l'ho anche con la stampa specializzata "europea" che ha sempre trascurato lo "standard giapponese" lasciando impantanati i pochi utenti che hanno creduto e continuano a credere nell'MSX...
E più o meno lo stesso è successo al Commodore 16 grazie alla politica della casa americana che per riempire il vuoto tra il C64 e l'uscita dell'Amiga ha tirato fuori prima il 128 e poi C16 e PLUS4 per poi abbandonarli a se stessi...
GRRRRR... che rabbia!
Credo sia quello che pensano quasi tutti gli utenti di queste stupende macchine ormai rimasti soli, pochi ma buoni...
Buon divertimento!

LA REDAZIONE

sfida al co

1. POLDO

Approfittando della tua assenza da casa per combattere durante l'ultima spedizione nel Lontano Oriente, il tuo eterno rivale McCornick si è impadronito della tua casa, di tutti i tuoi avere e, soprattutto, della tua legittima moglie, facendone la sua schiava.
Al tuo rientro, dopo anni di aspri combattimenti, vissuti in mezzo ai pericoli, hai quindi l'amara sorpresa di ritrovare tutto quanto avevi di più caro usurpato dal perfido McCornick, che spadroneggia per casa tua con aria tracotante.
Come fare per rientrare in possesso di ciò che ti spetta? L'impresa si presenta veramente ardua, ma i duri anni di guerra hanno irrobustito il tuo corpo e rafforzato il tuo spirito, e tu non sei certo tipo da tirarti indietro quando si tratta di combattere per la giustizia, per il tuo paese o per te stesso. Una volta caricato il programma premi la barra di spazio per iniziare, quindi collega il joystick in porta 2. In basso vedrai visulizzato un menù di gioco con queste opzioni:

GIOCATORI	1 o 2
LIVELLO	da 0 a 4
VITE	da 4 a 9
QUADRO	da 0 a 5
INIZIA	per cominciare la partita.

Per modificare le selezioni delle varie opzioni muovete verticalmente la leva del joystick fino a portare la barra lampeggiante sul punto desiderato e premete fire per variare il numero indicato. Per iniziare la partita, portate la barra su INIZIA e premete Fire due volte.
Il vostro personaggio parte dall'angolo destro in alto; per riuscire a sconfiggere McCornick devi dimostrare di essere il legittimo possessore dei tuoi beni (nonché il legittimo marito di tua moglie!) raccogliendo in successione le lettere P-O-L-D-O che compongono il tuo nome.

POLDO

Quando ti trovi in difficoltà premi Fire per entrare nelle porte. Quando hai perso una vita, premi Fire per ripartire.

2. DIABLO

Eccovi un gioco che non esitiamo a definire psichedelico per i mirabolanti effetti di colori, suoni e luci che appariranno sul teleschermo o sul monitor.
Siamo nell'anno 20001 e ancora una volta la Terra, diventata senza dubbio il più florido e ricco pianeta della Galassia e quindi preda di ogni genere di pirati e avventurieri, è nel mirino dei corsari dello spazio, una razza guerriera e vagabonda che vive depredando le ricchezze delle civiltà più pacifiche ed evolute.
Ma nessuno al comando militare aveva previsto un così repentino e violento attacco e che le forze nemiche si sarebbero organizzate così presto in un attacco in massa contro la base missilistica di cui tu sei comandante e a cui è affidata la difesa di tutto l'emisfero boreale della Terra.
I corsari dello spazio utilizzano strane ma micidiali astronavi d'assalto, estremamente veloci e manovrabili e quindi difficilissime da colpire.
Premi Fire per attivare l'armamento della base e collega il joystick in porta 2 per comandare il complesso sistema di puntamento.
A causa della mobilità delle astronavi nemiche, le uniche armi che si sono rivelate efficaci sono i collimatori a barriera, in grado di spazzare rapidamente nelle due direzioni il campo di battaglia, facendo piazza pulita degli avversari.
Ma attenzione!
I collimatori hanno un'autonomia limitata e quindi tenete d'occhio l'indicatore del tempo rimanente, posto a destra in alto.
Coraggio, datevi da fare!
Avete solo tre possibilità prima di dovervi dare alla fuga!

3. GOLF

Si racconta che il golf sia stato inventato nel 1604 da Robert Robertson.
Si tratta di un gioco molto popolare nella jet society e spesso sinonimo di appartenenza agli alti ceti sociali per gli alti costi d'iscrizione e di frequenza ai vari campi e club.
Il golf è concettualmente semplicissimo, ma in realtà richiede una tecnica raffinata, una perfetta condizione fisica e un buon allenamento.
Il gioco consiste nel mandare in una buca una piccola palla dalla caratteristica superficie a buccia d'arancia.
La palla viene colpita con un'apposita mazza, detta club, partendo da una piazzola fino ad arrivare al green, dove si trova una piccola buca, visibile anche a distanza grazie ad una bandiera.
Scopo del gioco è quindi di mandare la palla in buca col minor numero di colpi; ogni partita si disputa su un percorso composto da 9 o 18 buche e ovviamente vince chi alla fine ha usato il minor numero di colpi.
Per rendere più vario e appassionante il gioco, i percorsi sono disegnati in modo da frapporre vari ostacoli naturali e artificiali tra piazzola di partenza e la buca: terreni accidentati, laghetti, alberi e cespugli rendono indispensabile studiare accuratamente ogni colpo per evitare figuracce!
Ma non temete, il C16 e il Plus/4 vi permetteranno di allenarvi comodamente a casa vostra, studiando con calma le tecniche e le regole di gioco.
Premete F1 per iniziare la partita. Potete giocare da soli o con un vostro amico, su un campo a 9 o a 18 buche.
Per tirare, premete la barra di spazio, regolate la forza del colpo con i tasti cursore laterali, quindi premete Return, poi usate nuovamente i tasti cursore per decidere la direzione e premete Return per tirare.

4. POKER

Da sempre il poker è considerato il re dei giochi d'azzardo, e anche nel cinema abbiamo visto innumerevoli partite, da tempi dei film western dove i mazzi di carte avevano sempre almeno cinque assi. Le regole del gioco sono abbastanza semplici: si distribuiscono a ogni giocatore 5 carte, coperte, con cui bisogna cercare di costruire una delle combinazioni vincenti che vedremo tra poco.
Se le carte non sono buone, è possibile cambiarle, fino a quattro, ovviamente una volta sola, dopodiché si aprono le scommesse ed inizia il vero gioco d'azzardo, che consiste nel cercare di capire se la propria combinazione batte o meno quella degli avversari, regolando di conseguenza le proprie puntate. La versione da casinò è un po' diversa, in quanto si gioca contro il banco puntando in anticipo e le vincite dipendono dal tipo di combinazione ottenuta:

Coppia vestita (almendo 2 fanti): alla pari
Doppia coppia (2 coppie di carte): 2 a 1
Tris (3 carte uguali): 2 a 1
Scala (5 carte di valore crescente): 4 a 1
Colore (5 carte dello stesso seme): 5 a 1
Full (1 tris e una coppia): 7 a 1
Poker (4 carte uguali): 20 a 1
Scala reale (5 carte in scala dello stesso seme): 50 a 1
Scala reale con asso (la combinazione più forte!): 250 a 1.
Per giocare scegliete il valore dei gettoni, premendo un tasto da 1 a 4, quindi premete il tasto funzione F7 per iniziare la partita.
Per puntare i gettoni premete Return, e spazio per giocare la mano.
Selezionate le carte da tenere premendo i tasti corrispondenti, dall'1 al 5: le altre verranno cambiate, quindi se avrete realizzato una combinazione vincente ricevere la vincita.
Buona fortuna!

5. CRAZY TRAIN

Ed eccovi adesso nei panni di un ferroviere, col compito di trasportare merci e persone in un intricato sistema ferroviario.
Se le ferrovie ai tempi del West erano costituite da un solo binario che si snodava per miglia e miglia da una costa all'altra, il progresso e le esigenze della società moderna hanno cambiato enormemente i sistemi di comunicazione e di trasporto.
Infatti, la sala di controllo di una moderna stazione ferroviaria contiene decine di potenti computers, monitor e sofisticati sistemi di sicurezza per la prevenzione degli incidenti, per poter smistare il traffico nel miglior modo possibile.
Volete provare a cimentarvi in questa non facile opera?
Per iniziare, caricate il programma e scegliete il livello di difficoltà con i tasti cursori laterali (vi raccomandiamo di partire con un livello 1 o 2), quindi premete spazio.
Fate attenzione soprattutto agli scambi e ai binari morti, per non fare deragliare il vostro treno.
Cercate di trovare e agganciare i vagoni disposti qua e là, per portare il massimo carico.
Per muover il treno, usate i tasti cursore verso sinistra o destra, oppure i tasti ‹ e ›, mentre la barra spazio permette di frenare.
La velocità massima del convoglio è di 60 chilometri all'ora.
Sulla sinistra è indicato il carburante, in percentuale rispetto al pieno; quando sta per esaurirsi, portatevi presso uno dei punti di rifornimento, indicati da una F.
Per manovrare gli scambi premete il tasto P o HELP.

6. BATTLESHIP

Le forze aliene hanno invaso il pianeta di Zork e si stanno preparando a sferrare l'attacco finale col quale contano di distruggere ogni forma di vita sul pianeta.
Qualcuno però è intervenuto a fermare l'avanzata nemica. Questa persona... sei tu!
Molto eroicamente sei salito a bordo della tua piccola ma veloce astronave da battaglia e tenti il tutto per tutto per salvare il futuro del pianeta.
Le forze nemiche hanno mandato la loro ammiraglia all'attacco: un potentissimo incrociatore astrale dotato di un cannone laser a tiro rapido che non perdona, capace di annientare con un colpo solo la tua astronave.
L'incrociatore nemico è formidabilmente corazzato, e l'unico punto debole è costituito dalla torretta del cannone laser, che però è tutt'altro che facile da colpire in quanto continua a spostarsi da un capo all'altro dell'incrociatore.
Per riuscire a colpirlo, devi prima aprirti un varco nella spessa corazzatura, evitando nel frattempo i colpi del laser nemico che fa fuoco in continuazione.
Per giocare, collega il joystick in porta 1 e premi Fire per giocare. Per arrestare temporaneamente il gioco puoi premere il tasto P; premi nuovamente P per ripartire.
Hai tre vite a disposizione, sullo schermo vedrai indicati, da sinistra a destra, il punteggio realizzato, il record del giorno e le vite ancora a disposizione per portare a termine la distruzione dell'incrociatore nemico.

DISEGNA CARATTERI

Come molti di voi sapranno già, i computer considerano i caratteri alfanumerici semplicemente come dei numeri, generalmente da 0 a 255, cioè a 8 bits.
Esiste poi un'apposita parte del sistema operativo che associa ad ogni valore numerico un carattere, secondo una determinata codifica: l'insieme di queste corrispondenze si chiama SET DI CARATTERI, e da esso dipende la visualizzazione delle informazioni sullo schermo video.
All'interno del nostro computer sono presenti due set di caratteri, uno contenente i caratteri maiuscoli e i segni grafici, l'altro comprendente sia le maiuscole che le minuscole, ma con un numero ridotto di caratteri grafici.
La selezione tra i due set viene fatta premendo contemporaneamente i tasti SHIFT e C=.
Tramite questo listato potrete visualizzare o stampare sia la matrice di 8*8 pixel di ogni carattere, sia la sua codifica in decimale e in binario, dove a ogni bit corrispone un pixel.
Dopo aver dato il RUN, specificate se volete usare anche la stampante, quindi scegliete il set desiderato e digitate il numero del carattere che volete visualizzare.

```
100 POKE 55,255:POKE 56,54:CLR:COLOR 0,1:COLOR 4,1:SCNCLR
110 PRINT "■M┌▣▣▣▣▣┬ D000 D7FF 3700▣▣X▣▣G┌130"
120 POKE 239,5:POKE 1319,19:FOR T=0 TO 3:POKE 1320+T,13:NEXT:END
130 PRINT "♥▣▣RISULTATI SU STAMPANTE? (S/N) ";:GETKEY R$:PRINT R$
140 PRINT:IF R$="N" THEN 170
150 OPEN 4,4:PRINT#4
160 PRINT#4,"***STAMPA IMMAGINE CARATTERI***" :PRINT#4
170 D$ = "":PRINT "▣QUALE SET VUOI USARE: "
180 PRINT "  1  PER SET MAIUSCOLO/GRAFICO"
190 PRINT "  2  PER SET MAIUSCOLO/MINUSCOLO"
200 INPUT"▣▣ OPZIONE:";D$:IF D$="" THEN 500
210 IF D$ <> "1" AND D$ <> "2" THEN PRINT"▭▭▭▭":GOTO 200
220 M$=" SET MAIUSCOLO/MINUSCOLO"
230 IF D$="1" THEN M$=" SET MAIUSCOLO/GRAFICO"
240 PRINT "▣▣SCRIVI IL CODICE DEL CARATTERE"
250 PRINT "IN DECIMALE";:SW=0:INPUT D:PRINT
260 IF D<0 OR D>255 THEN PRINT"▭▭▭▭▭▭":GOTO 220
270 D1=D:IF D>127 THEN D=D-128:SW=1
280 A=14080+(VAL(D$)-1)*1024
290 FOR K=0 TO 7:D(K)=PEEK(A+D*8+K):NEXT K
300 PRINT "♥▣▣D/CODE=";D;M$:PRINT
310 PRINT "CARATTERE CORRISP. AGLI 8 BYTES:":PRINT
```

```
320 PRINT "CARATTERE  VAL. BINARIO  VAL. DEC.":PRINT"■":IF R$="N" THEN 380
330 PRINT#4,"D/CODE=";D1;M$:PRINT#4
340 PRINT#4,"CARATTERE CORRISP. AGLI 8 BYTES:"
350 PRINT#4
360 PRINT#4,"  CARATTERE    VAL. BINARIO   ";
370 PRINT#4,"VAL. DEC.":PRINT#4
380 FOR K=0 TO 7:D$(K)="":E$(K)="":B=D(K)
390 FOR J=0 TO 7:IF INT(B/2↑(7-J))=0 THEN 430
400 IF SW=1 THEN D$(K)=D$(K)+" ":E$(K)=E$(K)+"0":GOTO 420
410 D$(K)=D$(K)+"*":E$(K)=E$(K)+"1"
420 B=B-2↑(7-J):GOTO 450
430 IF SW=1 THEN D$(K)=D$(K)+"*":E$(K)=E$(K)+"1":GOTO 450
440 D$(K)=D$(K)+" ":E$(K)=E$(K)+"0"
450 NEXT J:NEXT K:FOR K=0 TO 7
460 PRINTD$(K);"      ";E$(K);"       ";D(K)
470 IF R$ = "N" THEN 490
480 PRINT#4," ";D$(K);"          ";:PRINT#4,E$(K);"       ";D(K)
490 NEXT K
500 IF R$="S" THEN PRINT#4: GOTO 170: ELSE 170
```

MSX ch

1. DIGGER

Sotto la superficie terrestre si annidava antichi mostri cheminacciavano il popolo dei Chaks, abitanti della Terra ormai dal 2042, data in cui il nostro pianeta tornato a fiorire dopo la distruzione nucleare avvenuta alla file del XX secolor. I Chaks arrivarono da Urano con l'intenzione di colonizzare il nostro pianeta e, come spesso accade quando si va ad abitare in una nuova casa, cercarono subito di mettere le cose a posto rendendo più vivibile il luogo. Per prima cosa tentarono di eliminare i mostri dimoranti sotto la superficie terrestre perché, un giorno, avrebbe potuto venire "alla luce" e creare problemi. Drake, il droide con la maggiore esperienza in questo settore, venne incaricato della terminazione. Il suo compito consisteva nel penetrare nel sottosuolo alla ricerca dei pericolosi animali. Per eliminarli poteva disporre di uno speciale arpione che, una volta colpiti i nemici, li gonfiava al punto di farli scoppiare.
Nella ricostruzione simultata su computer della mitica impresa di Drake, tu coprirai il ruolo dell'eroico protagonista. Naturalmente, una volta eliminati gli "ospiti" indesiderati di ogni quadro, dovrai far "pulizia" anche nel successivo e, naturalmente, proseguendo nell'azione di gioco, il tuo compito si farà sempre più arduo. In ogni schermo potrai disporre, oltre all'arpione, di grossi massi che potrai usare per schiacciare i mostriciattoli che ti inseguono. Per far cadere i massi basterà scavare un buco sotto di loro: essi crolleranno facendo piazza pulita di tutto ciò che si trova sotto di loro, ma attento a non farti travolgere!

COMANDI
Tasti:
[CURSORI] = Movimenti
[SPAZIO] = Inizio gioco / Fuoco
Joystick in porta 1

2. POYAN

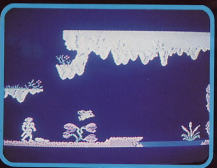
Forse non tutti ricordano un gioco che, personalmente, mi ha fatto passare ore spensierate in un bar davanti ad un videogioco diversi anni fa. Questo gioco è uno di quelli in cui ho investito più monete da duecento lire. Come potete vedere nella foto, nella parte destra dello schermo c'è una cabina tipo ascensore in cui trova posto un simpatico maialino di cui vestite i panni. Il maialino, che si può spostare in su ed in giù grazie all'aiuto di altri suini, deve fare in modo che i lupi, dotati di innumerevoli palloncini, raggiungano la cima della montagna e spingano tutti insieme la grossa roccia proprio sulla vostra testa. Per impedire ai lupi di arrivare in alto avete a disposizione un arco e un'illimitato numero di frecce con cui dovete cercare di colpire i palloncini dei lupi. Quando il numero di lupi è sufficiente a spingere la roccia non avete più scampo. Quando cresce la difficoltà del gioco a seconda del livello raggiunto, i lupi potranno insediarsi anche sulla scala posta alla vostra destra tentando di azzannarvi il popò.
Tra i diversi livelli del gioco troverete diversi livelli-bonus in cui dovrete colpire e affondare il maggior numero possibile di palloncini.
Prima di poter giocare dovrete selezionare, tramite il joystick o i tasti del cursore, il numero di giocatori nel menù principale. Una volta scelto il numero di giocatori (lampeggiante) potrete iniziare il gioco premendo lo spazio o il tasto di fuoco del joystick. Dopo questa selezione dovrete "subire" per alcuni secondi la musichetta di presentazione e quindi potrete giocare.
Nella parte superiore della corda potrete vedere una mega-freccia bonus che, se raccolta, potenzia notevolmente il vostro arco consentendovi di abbattere più lupi contemporaneamente.

COMANDI
Tasti:
[CURSORI] = Movimenti cabina & Scelta opzione
[SPAZIO] = Fuoco & Selezione opzione
Joystick in porta 1

3. CAMELOT

Eccoci di nuovo con un gioco di tipo Arcade/Avventure in cui dovrete peregrinare tra numerosissimi schermi, tutti diversi tra loro.
Il gioco è ambientato ai tempi di Re Artù e voi non potete essere che un cavaliere della Tavola Rotonda alla ricerca del Sacro Graal, la coppa in cui bevve Cristo tanti, tanti anni fa. Così, alla ricerca del Graal, dovrete affrontare folletti, trolls, maghi e fattucchiere senza armi speciali (questa non è fantascienza!) ma solo con la spada "standard in dotazione" al cavaliere medioevale.
L'unica cosa che potrete fare è di saltare in lungo sia per evitare i mostri, decisamente deleteri, che per cambiare schermo e passare da un piano all'altro. Non resta altro che raccogliere tutto il coraggio che si dispone e partire per questa impresa, forti delle nove vite che vi vengono messe a disposizione.
Quando il caricamento del programma termina sullo schermo appare velocemente la presentazione iniziale e quindi si passa al menù principale che offre due sole opzioni:

1 - CURSORI
2 - JOYSTICK.

Per selezionare una delle due opzioni sarà sufficiente premere il tasto numerico corrispondente. Selezionando il tipo di controllo farete anche iniziare il gioco.

COMANDI
Tasti:
[CURSORI] = Movimenti & Salto
[SPAZIO] = Usa Spada
Joystick in porta 1

allenge

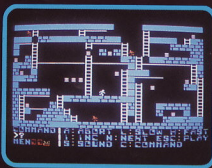
4. RUNNER

Lord Hell ha rubato i tesori che costituiscono il prestigioso patrimonio del tuo casato e li ha nascosti nelle segrete del suo palazzo. Il tuo obbiettivo è di penetrare all'interno delle segrete, recuperare il tesoro, sfuggire alle Guardie del Male e... uscirne illeso! Devi raccogliere tutti gli oggetti presenti sullo schermo utilizzando poi scale e funi per la fuga. Sentendo gli inseguitori alle costole e non trovando vie di scampo, puoi scavare una buca tra i mattoni grazie al tuo potente acido e farvi cadere dentro i tuoi avversari. Una volta recuperati tutti gli offetti presenti in un determinato quadro di gioco, apparirà sullo schermo una scala che costituisce la tua via d'accesso verso un nuovo livello di gioco. Inoltre, è sempre presente il rischio di finire in una trappola inizialmente preparata per i tuoi inseguitori.
In questo classico gioco tipo "platform" hai la possibilità di modificare gli schermi dei diversi livelli di gioco presenti grazie all'editor interno che si attiva tramite il tasto [ESC]. Ma ecco i numerosi tasti da utilizzare:

COMANDI
Tasti FASE DI GIOCO:

[CURSORI] = Movimenti
[Z] = Scava buca a sinistra
[X] = Scava buca a destra
[ESC] = Editor/Ritorno al gioco
[A] = Resetta il gioco dallo schermo in cui ti trovi
[L] = Rallenta velocità gioco
[H] = Autementa velocità gioco
[N] = Passa allo schermo successivo
[S] = Attiva/Disattiva Sonoro
[I] = Incrementa il numero di vite disposizione
[C] = Passaggio ai comandi di Edit

Tasti MODO EDIT:
[R] = Numero dello schermo da modificare (1 -85)
[CURSORI] = Seleziona schermo
[E] = Accesso allo schermo selezionato. In questa fase con i tasti del cursore puoi spostarti sui pezzi da utilizzare premendo il tasto numerico corrispondente.
[P] = Per giocare sullo schermo selezionato

Tasti FASE POST-EDITOR:
[M] = Memorizza lo schermo disegnato
[S] = Salva lo schermo su cassetta. E' necessario dare un nome al programma e poi inserire una nuova cassetta nel registratore, per poi proseguire come in una normale operazione di SAVE.
[L] = Carica da cassetta il file col nome indicato.
[C] = Cancella lo schermo disegnato
[R] = Ritorna al quadro di gioco al punto in cui era stato interrotto

5. MAGICAL

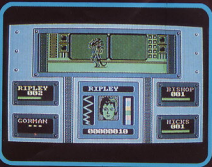
Da un po' di tempo a questa parte succedono cose strane a Wiz. Potremmo addirittura affermare che succedono cose più strane del solito. Dovete sapere che Wiz è un luogo abitato da maghi, streghe, fattucchiere, fantasmi, creature magiche e così via. A dire la verità, fino ad ora, nessuno degli abitanti intorno a Wiz si era mai molto preoccupato di quello che accadeva nella vasta zona sviluppatasi vicino alle loro case e nell'adiacente castello, ovviamente infestato da strane creature.
Il punto è che se, fino ad oggi, questi "coinquilini" se ne erano stati "buoni", ora sembra che si siano incattiviti e che non riescano più a vivere in armonia tra di loro. Qualche cosa di terribile deve essere successo in quel maledetto castello.
Il tuo compito sarà proprio quello di scoprire cosa può essere accaduto a tutti questi maghi e a queste - una volta tranquille creature che vivono a Wiz. C'era bisogno di qualcuno che andasse a mettere le cose a posto e per questo sei stato scelto proprio tu perché amico di un mago che ti ha insegnato delle piccole cose che potrebbero esseti utili una volta giunto a Wiz.
Innanzitutto dovrai vestirti come un fantasma, questo per confonder loro le idee e, poi, usando i tuoi poteri, eliminarli lanciandogli contro degli incantesimi. Dovrai stare ben attento a non toccarli perché perderesti immediatamente una delle tue tre vite a disposizione. Entrerai nella zona armato di un bastone magico, lo vedrai visualizzato sullo schermo, in basso, circondato da un riquadro rosso. Userai questo bastone per lanciare degli incantesimi mortali a tutti i fantasmi colorati di rosa che, se eliminati, ti daranno via libera.
Attento però agli altri fantasmi blu: sfuggili, perché il tuo bastine nulla potrà contro la loro foza. Attento poi ai nemici rossi che ti lanceranno contro degli incantesimi mortali. Potrai evitarli solo saltando da qualche altra parte o comunque spostandoti. Inoltre potresti anche essere colpito da fiocchi volanti viola-celesti che, oltre a cambiare il tuo colore, rallenteranno i tuoi movimenti, oppure da un pugno d'acciaio che passerà spazzando via tutto. Questo non ti farà perdere una vita se sei in alto, la perderai invece se non riuscirai ad evitarlo quando passerà in basso. Lungo il tuo cammino incontrerai anche dei bauli chiusi che potrai aprire lanciando un incantesimo col tuo bastone. Una volta colpito, il baule si aprirà e verrà fuori un oggetto magico che dovrai raccogliere. Gli elementi magici raccolti appariranno sullo schermo, in basso, vicino al bastone. Questi oggetti saranno: un boomerang, una stella magica, una pozione, il fuoco, una ampolla, un topolino volante, una clessidra e una gemma che, però, potrai raccogliere solo se riuscirai ad entrare nel castello. Una volta recuperati gli elementi magici, potrai attivarli semplicemente spostando il joystick verso di te. Ricorda però che potrai usarli una sola volta!

Ancora una cosa: il tempo è limitato ed è indicato sullo schermo in alto a destra. Ricorda che una volta a Wiz solo la tua astuzia e la tua abilità potranno aiutarti a uscirne vivo!

COMANDI
Tasti:
[CURSORI] = Movimenti & Salto
[SPAZIO] = Inizio gioco / Fuoco
Joystick in porta 1

6. ALIENS

Questo gioco di fantascienza prende spunto dalla battaglia contro gli alieni condotta dai sei membri dell'equipaggio di una missione esplorativa. Lo scopo consisterà quindi nell'ispezionare al base alla ricerca degli alieni che devono essere distrutti prima che possano riprodursi. L'obiettivo finale sarà quello di raggiungere la regina dei mostri alieni per ucciderla e distruggere con essa tutte le uova. Durante il gioco sarà possibile selezionare uno dei sei componenti l'equipaggio della missione.
Nella parte superiore dello schermo verrà visualizzato il membro dell'equipaggio selezionato mentre, nella parte bassa, potrai vedere la sua foto, la situazione metabolica del corpo, la carica di energia del laser, il nome e le condizioni fisiche di ogni personaggio e, infine, il punteggio raggiunto.
Sotto il nome di di ogni uomo vedrai evidenziato un numero indicante la vicinanza degli alieni. Il valore più basso è uno. Quando questo indice salirà, vorrà dire che ti stai avvicinando ad un alieno. Il valore quattro indicherà la presenza dell'alieno nella stessa stanza in cui ti trovi anche tu. Per cambiare zona della base basterà posizionare il mirino su una porta qualsiasi e premere il tasto di Invio.
Per controllare gli spostamenti del mirino utilizza i tasti del cursore. Per selezionare il tuo uomo dovrai invece selezionare il tasto corrispondente all'iniziale del suo nome. Per l'attivazione del personaggio Bishop premi [I]. Entrando in pausa potrai vedere visualizzata su schermo la situazione in corso della tua misssione, il numero dei nemici abbattuti, le condizioni dell'equipaggio e via dicendo.

COMANDI
Tasti:
[CURSORI] = Movimenti
[SPAZIO] = Inizio gioco / Fuoco
[RETURN] = Passa attraverso la porta selezionata col mirino
[M] = Resetta il gioco
[P] = Attiva/Disattiva la pausa. Durante la pausa verrà visualizzata la situazione della battaglia.
[B] = Seleziona BURKE
[R] = Seleziona RIPLEY
[V] = Seleziona VELASQUEZ
[G] = Seleziona GORMAN
[H] = Seleziona HICKS
[I] = Seleziona BISHOP

7. L'ISOLA

Ecco una nuova classica avventura di testo. Per poter utilizzare l'avventura non dovete usare il solito comando LOAD"cas:",R oppure RUN-"cas:", ma dovete usare l'istruzione CLOAD. Quando il caricamento è terminato dovete digitare RUN e premere il tasto di invio.

COME GIOCARE
Giocare una adventure significa guidare il protagonista di una storia verso uno scopo ben preciso, impartendo comandi tipo "PRENDI IL MARTELLO" o "ROMPI IL VASO" formate essenzialmente da un verbo seguito da un sostantivo e dalla pressione del tasto di immissione [RETURN]. Le istruzioni vanno impartite per esteso, in lingua italiana e utilizzando la seconda persona singolare. Potrete spostarvi nelle quattro direzioni cardinali NORD, SUD, EST, OVEST e ordinare SALI e SCENDI o, se la situazione lo richiede, ATTRAVERSA, ENTRA o ESCI. La descrizione del luogo in cui vi trovate e degli oggetti o dei personaggi presenti potrà scomparire a causa dello spostamento verso l'alto (SCROLL verticale) del testo : per rivederla basterà digitare DESCRIVI o GUARDA. Gli oggetti potranno essere manipolati per mezzo dei verbi PRENDI, POSA e INDOSSA o TOGLI (questi ultimi due se si tratta di ornamenti o di vestiti). Per inventariare gli oggetti trasportati dovrete digitare LIST oppure INVENTARIO. L'esame di oggetti, luoghi o personaggi attraverso ESAMINA potrà condurre in molti casi alla scoperta di altri oggetti o di particolari nascosti : se ciò dovesse accadere, lo schermo verrà cancellato e la descrizione del luogo aggiornata. Infine, ricordandovi che la soluzione di una avventura può richiedere anche giorni o settimane di "duro impegno", tenete presente che è possibile registrare su un'altra cassetta disco la situazione corrente di gioco col comando SAVE e ripristinarla in qualsiasi momento col comando LOAD. In seguito a tali istruzioni vi verrà chiesto di inserire la cassetta di destinazione dei dati e premere un tasto, dopodiché lo schermo si cancellerà per il tempo occorrente alla lettura o alla scrittura dei dati. Questo è il vocabolario essenziale di ogni avventura che si rispetti, ma il Dizionario di Gioco (PARSER), contiene molti altri vocaboli, relativi alle situazioni che incontrerete, per cui impartite normalmente le vostre istruzioni, cercando di cambiare i vocaboli nel caso in cui il computer non vi comprendesse. Ora però leggete attentamente il seguente racconto, prassi assolutamente indispensabile per poter affrontare e risolvere l'avventura.

PROLOGO
Trevor Scott era il classico "Avventuriero Solitario" sempre a caccia di curiosità archeologiche ed oggetti preziosi del passato: aveva sempre affrontato da solo i pericoli del suo "mestiere", che svolgeva in proprio o per commissione, ormai da diversi anni. Ma la faccenda della "Maschera d'Oro" di Alak Khal aveva tutta l'aria di presentarsi abbastanza spinosa, per cui si era dato da fare per cercarsi un valido collaboratore fra gli squattrinati che frequentavano la Taverna del Porto.
Era stato proprio in quella taverna che, alcune sere prima, aveva ascoltato la storia della "Maschera d'Oro" da un vecchio marinaio alcolizzato, e gli era sembrata abbastanza interessante da perderci un po' di tempo e di denaro.
Tra i clienti del locale ve n'era uno che Trevor aveva già avuto come collaboratore alcuni prima: si trattava di Boris Ruskin, un marinaio russo dal passato oscuro e dal futuro incerto. Trevor non si era mai fidato fino in fondo del suo vecchio socio Boris, nemmeno quando affrontò insieme a lui una pericolosissima spedizione archeologica in Colombia; e da quando gli aveva proposto di collaborare con lui alla spedizione per il ritrovamento della Maschera d'Oro sull'atollo di Alak Khal, il livello di fiducia stava calando rapidamente a causa dello strano atteggiamento del russo, che stava tra l'interesse e lo scetticismo.
Non poteva poi dimenticare la risata sarcastica del suo socio quando gli aveva confidato che il nome dell'isola avrebbe aperto, secondo la leggenda narrata dal vecchio, il cuore dell'isola stessa per giungere fino alla Maschera. Ma ora era troppo tardi per recriminare: erano già sopra l'isola col suo vecchio aereoplano, Boris alla guida, e fra qualche minuto avrebbero affrontato insieme i pericoli di quella piccolissima isola sconosciuta alle carte nautiche ufficiali. Fu mentre si preparava a prendere posto di fianco a Boris per l'atterraggio, che Trevor si trovò puntata addosso un'automatica presumibilmente - pensò - carica.
L'arma era saldamente sostenuta dalla mano sinistra di Boris Ruskin. "Prendi quel paracadute e buttati, prima che cambi idea e decida di darti in pasto ai pesci, socio!", tuonò Boris col suo tipico accento sovietico.
"Cosa ti prende, fratello, vuoi farmi uno scherzo?" - Trevor si sforzò di sorridere, ma ne uscì solo una smorfia.
Il russo sollevò la pistola mirando in mezzo agli occhi, e con un sibilo ripetè rabbiosamente il suo ordine: "Ti ho detto di buttarti! Non avrai mica creduto che avrei rischiato la pelle con un pazzo come te selo per dar retta alle fantasie di un vecchio alcolizzato?!?" -scoppiò in una risata - "Ho fatto finta di stare al gioco solo per sbarazzarmi di te in modo pulito e perfetto, per poi vendere la merce del tuo magazzino e ricavare qualcosa anche da questa vecchia carretta volante!".
Prima di lanciarsi nel vuoto Trevor guardò un'ultima volta negli occhi il suo socio ed urlò: "Attento Boris, potrei sopravvivere e ritrovarti...e allora...".
Non finì la frase: il proiettile che gli passò sibilando vicino all'orecchio lo convinse a lasciarsi andare nel vuoto oltre il portello... pochi minuti dopo era già sull'isola, al centro di una radura e il suo vecchio bimotore era ormai scomparso... era solo, senza armi e senza mappa: ma si fidava del suo istinto, e pensò che, con un po' di fortuna, avrebbe anche potuto trovare la Maschera d'Oro e tornare alla civiltà... forse!

BASIC (Parte 5)

Le istruzioni di controllo (salto, salto condizionato, esecuzione condizionale, ecc.) consentono di alterare la sequenza di svolgimento del programma, a seconda delle condizioni che si presentano. Sono le istruzioni che danno "intelligenza" al computer.
Nei diagrammi di flusso abbiamo visto come il percorso logico del programma non sia quasi mai costituito da un'unica linea, ma possa seguire diverse strade, con diramazioni e ritorni. Introduciamo adesso le principali istruzioni BASIC che consentono di realizzare questo controllo della sequenza di esecuzione delle istruzioni.

SALTO INCONDIZIONATO - Istruzione GOTO

Osservate questo programma:

```
10 PRINT "CIAO! IO SONO UN COMPUTER."
20 GOTO 10
```

La linea 10 fa stampare sullo schermo la stringa racchiusa tra virgolette (qualunque cosa ci sia scritto). La linea 20 dice: GOTO 10. "Go to" in inglese significa "vai a" ed in questo caso dice al computer: Invece di eseguire la linea successiva (con numero maggiore di 20), vai alla linea 10, e poi prosegui da lì. Come risultato, il programma "torna" alla liena 10, la esegue (stampando), prosegue alla 20, viene rimandato alla 10, e continua a stampare la stringa all'infinito. Questo è un esempio di LOOP (ciclo o anello chiuso), in cui il programma ripete (in questo caso per sempre) una certa sequenza di istruzioni.
A proposito, per fermarlo bisogna premere i tasti [CTRL] e [C] (tenendo premuto il tasto [CTRL], premere [C]) oppure [CTRL] e [STOP] o [BREAK] o quello specificato dal manuale. Un programma arrestato in questo modo può essere fatto ripartito dallo stesso punto dove si era fermato scrivendo il comando CONT [RETURN] oppure CONTINUE [RETURN].

SALTO CONDIZIONATO - Istruzione IF...GOTO e IF...THEN

I calcolatori sono macchine stupide. I calcolatori sono macchine intelligenti. Entrambe le affermazioni sono vere, basta indendersi sul significato di "intelligenza". Il computer esegue senza capire, dunque è stupido, ma è in grado di prendere delle decisioni, quindi deve possedere una qualche forma di intelligenza. Questa non è altro, in realtà, che quella del programmatore, il quale gli ha spiegato in dettaglio (scrivendo il programma) come comportarsi in tutte le possibili situazioni di scelta.
Le istruzioni che effettuano queste scelte sono fondamentali: senza di esse il calcolatore eseguirebbe sempre la stessa sequenza di istruzioni e non sarebbe quella macchina potente e versatile che conosciamo. Vediamo subito in pratica un esempio di "intelligenza":

```
10 INPUT "SCRIVI UN NUMERO MINORE DI 9: ";N
20 IF N›=9 GOTO 10
30 PRINT "OK, VEDO CHE CI SIAMO CAPITI"
```

Cosa succede?
Il computer stampa il messaggio alla linea 10 e resta in attesa di un numero dalla tastiera. Quando lo riceve passa alla linea successiva (20). Alla linea 20 c'è scritto IF. In inglese, IF significa SE; il calcolatore si chiede SE il numero introdotto (N) è MAGGIORE(›) o UGUALE(=) a 9. Se questo è vero (quindi non è minore di 9), esegue il GOTO 10 che rimanda alla linea 10, dove viene chiesto di introdurre nuovamente il numero.
Se invece la CONDIZIONE (N maggiore o uguale a 9) non è verificata, allora non succede niente ed il programma prosegue alla linea seguente (30). Questo si chiama SALTO CONDIZIONATO: il programma "salta" a un'altra istruzione solo se la condizione è vera. Abbiamo insegnato al computer a "capire" quello che gli viene dato e ad accettare soltanto i numeri validi. In un certo senso, gli abbiamo dato un pò di "intelligenza".
Una volta erano pochissimi e ora sono ormai estinti i BASIC che consentono soltanto IF...GOTO. Tuttu i dialetti Basic esistenti permettono una forma più completa, che non consente soltanto il salto condizionato, ma anche l'ESECUZIONE CONDIZIONALE di una o più istruzioni.

Facciamo un esempio:

```
10 INPUT "DAMMI UN NUMERO: ";N
20 Q=N*N:REM CALCOLA QUADRATO
30 PRINT "QUANT'E' IL QUADRATO DI ";N
40 INPUT X:REM TENTATIVO
50 IF X ‹ Q THEN PRINT "TROPPO BASSO!":GOTO 30
60 IF X › Q THEN PROMT "TROPPO ALTO!":GOTO 30
70 PRINT "CENTRATO! ERA PROPRIO ";Q
```

Il programma chiede un numero (N) e ne calcola il quadrato (Q). Poi chiede un altro numero (X), che rappresenta un tentativo di indovinare il quadrato di N. A questo punto (linea 50) si verifica se il numero importato è minosre del valore da indovinare.
Se si, ALLORA (in inglese THEN) viene eseguito tutto il resto della linea 50. Se no, il resto della linea viene slatato e si prosegue alla linea 60.
Questo si chiama ESECUZIONE CONDIZIONALE di un'istruzione (nel nostro caso di due istruzioni, un PRINT ed un GOTO), in quanto le istruzioni vengono eseguite soltanto se la condizione è vera.
Il resto della linea 50 è composto da due istruzioni, separate

BASIC (Parte 5)

dal simbolo ":" (due punti). In BASIC due istruzioni separate da due punti sono eseguite una dopo l'altra, come se fossero su due linee successive (qualche BASIC usa un simbolo diverso e pochissimi, come quello dell'ormai estinto ZX81, non lo permettono affatto).
Dato che entrambe le istruzioni si trovano dopo il THEN, sulla stessa linea, o vengono eseguite entrambe (se la condizione è vera) o non ne viene eseguita nessuna (se la condizione è falsa).
L'effetto sul funzionamento del programma esempio è che, se il numero impostato (X) è minore della soluzione (Q), viene stampato un

```
"TROPPO BASSO!"
```

e viene poi eseguito un GOTO 30, che trona a richiedere un tentativo. Se il numero non è minore della soluzione (quindi è uguale o maggiore), il programma prosegue ignorando completamente il seguito della linea 50.
Alla linea 60, tutto come sopra, con la differenza che viene verificato se il numero è maggiore del risultato. Se è così, messaggio e ritorno alla linea 30. Infine, se il numero non è né maggiore né minore deve essere uguale: la risposta è esatta. Messaggio e fine del programma (linea 70).
Notate come la stessa istruzione PRINT possa stampare, ad esempio, una stringa (racchiusa tra virgolette) e, di seguito (punto e virgola), il valore di una variabile numerica.
Negli esempi di cui facciamo uso per illustrare il funzionamento delle istruzioni BASIC, useremo sempre nomi di variabili di una o due lettere al massimo (o una lettera ed un numero, come A3).
Questo perché in molti BASIC le variabili sono identificate soltanto dai primi due caratteri. E' sempre comunque utile inserire qualche REM che spieghi il signigicato delle variabili, come abbiamo fatto alla linea 40 (nelle altre linee le stringhe dei PRINT erano già abbastanza esplicative).
Per concludere, ormai tute le versioni di BASIC permettono anche la clausola ELSE (ALTRIMENTI), che specifica le istruzioni da eseguire se la condizione è falsa. Ad esempio:

```
IF A=3 THEN PRINT "A VALE 3" ELSE PRINT "NON VALE 3"
```

Questa costruzione è molto comoda perché se i gruppi IF-...THEN...ELSE sono NIDIFICABILI, cioè possono essere inseriti uno dentro l'altro (in genere un numero limitato di volte), consente una costruzione più "pulita" del programma. Ma qui stiamo sconfinadno dal BASIC verso i linguaggi strutturati come Pascal o C.

ESPRESSIONI LOGICHE

Un notevole passo avanti nella capacità di programmare si ha quando possiamo chiedere al calcolatore di prendere delle decisioni. Le "espressioni logiche", simile a quelle numeriche, permettono appunto al calcolatore di fare questa scelta. "Utility" e funzioni di "Debug" alleviano la fatica del programmatore e oggi non sono più patrimonio solo dei grandi calcolatori, ma anche dei piccoli personal e home computer. Grandezze numeriche e "stringhe di caratteri" sono i dati su cui può lavorare un personal computer. Abbiamo visto come nelle istruzioni condizionali, per esempio IF-...THEN, la scelta dipenda da una CONDIZIONE. In realtà la condizione può essere un'intera ESPRESSIONE LOGICA, il cui risultato viene usato ai fini della scelta. Un'espressione logica, come una normale espressione numerica, è composta da OPERANDI (variabili o costanti) separati da OPERATORI (per esempio, nell'espressione numerica A+3, A e 3 sono gli operandi e + è l'operatore). A differenza delle espressioni numeriche, pero', il risultato è un VALORE LOGICO, o BOOLEANO (dal nome del matematico George Boole), che può assumere due soli valori: vero o falso. Quindi, diamo una definizione più completa dell'istruzione di salto condizionato:

IF espressione logica GOTO numero linea

Se il risultato dell'ESPRESSIONE LOGICA è VERO, viene eseguito il GOTO; se è FALSO, l'esecuzione prosegue normalemnte con la linea successiva. Lo stesso vale, naturalmente, per gli altri tipi di espressione condizionale.
Nelle espressioni logiche si usano gli operatori relazionali e logici.

OPERATORI RELAZIONALI

Così detti perché verificano una relazione, accettano come operandi le normali variabili e costanti e restituiscono come risultato un valore logico.
Ecco un esempio:

```
IF A ‹3 THEN
```

Se A è minore di 3 l'espressione "A ‹3" è vera

Tra normali variabili numeriche, gli operatori relazionali hanno il consueto significato.
Ad esempio, 5 è maggiore di 3 (attenzione: -5 è minore di 3).
Tra variabili stringa, gli operatori = (uguale) e ‹› (diverso) verificano se le due stringhe coincidono carattere per carattere, spazi compresi.
Ecco un esempio tipico:

```
150 INPUT "ALTRI DATI ? ";A$
160 IF A$ = "NO" GOTO 200
170 IF A$ ‹› "SI" GOTO 150
180 ....
```

BASIC (Parte 5)

La linea 150 stampa

```
"ALTRI DATI?"
```

e aspetta che l'operatore scriva una linea, terminata con la pressione del tasto di invio, [RETURN] o [CR] . I caratteri scritti vanno a finire nella variabile stringa A$.
La linea 160 controlla se A$ è uguale a "NO", cioè se era stato scrittp "NO" sulla tastiera. Se è vero, salta alla linea 200.
La linea 170 è una "sicura" contro errori dell'operatore. Se non era "NO", doveva essere "SI". In caso contrario ritorna a chiedere l'INPUT.
Se A$ valeva "SI", il programma prosegue alla linea 180.
L'effetto degli altri operatori relazionali (maggiore, minore ecc.) su confronti tra variabili - e costanti - stringa non è uguale per tutti i BASIC. Di solito, comunque, viene effettuato un confronto alfabetico, fino alla lunghezza della stringa più corta. Questo permette di fare ordinamenti alfabetici, ricerche più veloci in un insieme di dati, ecc.

OPERATORI LOGICI

Capitano spesso situazioni più complesse di quelle risolvibili in una sola istruzione con gli operatori relazionali. Ad esempio, per sapere se il valore della variabile A è compreso tra 5 e 12 servirebbero due istruzioni:

```
330 IF A < 5 GOTO 350
340 IF A <= 12 GOTO 900:REM E' COMPRESO TRA
5 E 12
350 ...
```

L'istruzione alla linea 330 serve a saltare il secondo TEST (prova, controllo) nel caso in cui A sia minore di 5. Si tratta evidentemente di una forma scomoda e poco leggibile. Per fortuna, in quasi tutti i BASIC è possibile scrivere:

```
330 IF (A>=5) AND (A<=12) GOTO 900
```

che significa: se (è vero che) A è maggiore od uguale a 5, e (è anche vero che) A è minore od uguale a 12, allora vai alla linea 900.
AND ("e" in italiano) è un OPERATORE LOGICO. Gli operatori logici accettano come operandi due valori logici (nell'esempio i risultati dei due confronti) e restituiscono come risultato un valore logico. Gli operatori logici usati nel BASIC sono:

AND (PRODOTTO LOGICO, due o più operandi):
il risultato è vero solo se tutti gli operandi sono veri;

OR (SOMMA LOGICA, due o più operandi):
il risultato è vero se almeno uno degli operandi è vero;

NOT (INVERSIONE LOGICA, un solo operando):
il risultato è vero solo se l'operando è falso.

Nel nostro esempio, le parentesi erano probabilmente inutili, in quanto di solito in BASIC la precedenza (o priorita') degli operatori relazionali è maggiore di quella degli operatori logici. In altri termini, prima si eseguono i confronti e poi vari AND, OR, NOT. Un esempio di uso dell'operatore logico OR ("o" in italiano):

```
40 INPUT "DAI UN NUMERO TRA 2 E 18: ";N
50 IF (N < 2) OR (N > 18) GOTO 40
```

A differenza di altri linguaggi di programmazione, in BASIC non esistono VARIABILI LOGICHE (BOOLEANE) speciali, ma si usano le normali variabili numeriche. Una variabile ha valore logico FALSO se vale zero, VERO in tutti gli altri casi. Perciò è possibile scrivere:

```
40 IF A GOTO 100
```

in luogo di

```
40 IF A<> 0 GOTO 100
```

E' anche possibile assegnare ad una variabile numerica il risultato di un'espressione logica. Ad esempio:

```
120 R=(A=B)
```

Viene assegnato ad R il valore logico risultante dal confronto tra A e B (le parentesi sono per chiarezza). Ciò significa che R vale zero se A e B hanno valore diverso (confronto falso). Se A e B sono uguali, R assume il valore corrispondente al valore logico "vero".
In alcuni dialetti BASIC il valore di "vero" e 1 (uno), in altri è -1 (in ogni caso è diverso da zero e può essere usato in altre espressione logicha). Per il BASIC della nostra macchina il valore "vero" e -1 e se un dialetto BASIC usa -1 come valore logico per "vero" vuol dire che consente di manipolare i singoli bit della memoria. Non sviluppiamo ora l'argomento, come pure non parliamo per ora dell'uso avanzato delle espressioni logiche, per non creare inutile confusione. Meglio, prima, conoscere il BASIC "normale".

UTILITY

Nella maggior parte dei calcolatori, la semplice correzione di un carattere in una linea inserita in precedenza (tipica operazione di EDITING) è già un lavoro alquanto fastidioso. Ma le scomodità che il programmatore incontra aumen-

BASIC (Parte 5)

tano rapidamente con il crescere delle dimensioni del programma. Come faccio ad inserire 15 linee tra le linee 120 e 130? Ho già usato la variabile H3? C'è qualche GOTO alla linea 1380? Eccetera, eccettera.
Un programmatore professionista rischia di passare la maggior parte del proprio tempo risolvendo problemi del genere, invece di concentrarsi su una corretta scrittura (stesura, codifica) del programma.
Per questo esistono le UTILITY (funzioni o programmi di utilita'). Si tratta di funzioni già presenti insieme al BASIC (RESIDENTI) o che si possono introdurre leggendole da una memoria di massa (in genere un supporto magnetico o una ROM).
Possono essere state progettate dal costruttore del computer, da una SOFTWARE HOUSE (ditta di software) qualunque, o dal programmatore stesso.
Una volta inserite - se non sono già presenti - non aggiungono istruzioni BASIC, ma entrano far parte dei comandi disponibili in MODO DI EDITING (quando si scrive il programma, non quando gira) per rendere il lavoro del programmatore più comodo e veloce.

Le UTILITY INTEGRATE (già comprese) o disponibili per ogni singola macchina sono molto diverse per qualità e quantita'. Il primato spetta indubbiamente all'PC IBM, per il quale la varietà di software disponibile è immensa, seguito da APPLE e COMMODORE.
Poiché è impossibile descrivere, anche solo sommariamente, quello che il mercato offre, vediamo almeno i principali tipi di utility che possono facilitare il lavoro di chi scrive programmi e che già in parte si trovano nel BASIC della nostra macchina:

EDIT (MODIFICA)
Le utility di EDIT servono a rendere più comodo il lavoro di modifica delle linee già introdotte. Molto diverse da una macchina all'altra, sono particolarmente utili per i calcolatori che non dispongono del VIDEO EDITING (modifica diretta sul video).

AUTO-NUMBER (Numerazione automatica)
Già presente come comando in quasi tutti i BASIC, consente di introdurre automaticamente il numero della linea successiva all'ultima scritta, evitando errori e cancellazioni accidentali di altre linee. In genere è possibile specificare l'INCREMENTO, cioè di quanto aumentare ogni volta il numero di linea (di solito si usa 10). E' un'utility che può fare comodo, ma può anche creare intralci se non è ben realizzata (numeri di linea che appaiono anche se non voluti).

RENUMBER (RINUMERA)
Anche questa già presente a volte come comando, svolge una funzione importante: permette di RINUMERARE il programma od una sua parte, cioè di cambiare tutti i numeri di linea.
Di solito è possibile stabilire inizio e fine della zona interessata e PASSO (Incremento) dei nuovi numeri. Serve quando occorre aggiungere più linee già scritte. Oltre ai numeri di linea, vengono naturalmente cambiati tutti i RIFERIMENTI (GOTO, ecc.) alle linee stesse.

MOVE, COPY (SPOSTA, COPIA)
Permettono di spostare e/o duplicare gruppi di linee da un punto all'altro del programma. A volte sono incluse nel RENUMBER.

FIND, REPLACE (TROVA, SOSTITUISCI)
FIND serve a cercare un'istruzione, una variabile od una qualunque sequenza di caratteri usata nel programma.
REPLACE consente di rimpiazzarla con un'altra istruzione, variabile o sequenza specificata.

VARIABLE CROSS REFERENCE (RIFERIMENTO INCROCIATO DELLE VARIABILI)
Indica tutte le variabili usate nel programma, con linee nelle quali ciascuna è impiegata. Molto utile per risolvere problemi derivanti dall'impiego accidentale della stessa variabile con significati diversi.

LINE CROSS REFERENCE (RIFERIMENTO INCROCIATO DELLE LINEE)
Indica tutte le linee che sono riferite da altre linee (ad esempio linee destinazione di GOTO) e, per ciascuna di esse, le linee contenenti il riferimento. Di solito individua anche alcuni errori, come GOTO a linee inesistenti.

OPTIMIZER, CRUNCHER, COMPRESSOR (OTTIMIZZATORE, COMPRESSORE)
Toglie dal programma tutte le istruzioni di REM e le linee di soli REM, riducendo l'occupazione di memoria, aumentando la velocità di esecuzione e rendendo difficile la lettura del programma da parte di persone non autorizzate.
Le migliori utility di questo genere riducono ulteriormente le dimensioni del programma, riunendo quante più istruzioni possibile in una sola linea.
La lunghezza di un programma compresso puo; essere anche meno di metà dell'originale.

MERGE (FONDE, RIUNISCE)
Abbiamo già citato questa utility tra i possibili comandi BASIC. Serve a riunire due programmi in uno solo. Di solito, uno dei due è presente in memoria (centrale), mentre l'altro viene letto da un supporto magnetico (dischetto o nastro).
Consente di tenere da parte (IN LIBRERIA) una serie di parti di programma già pronte (ROUTINE), da inserire nel programma che si sta scrivendo.

BASIC (Parte 5)

Poiché l'elenco delle possibili utility potrebbe continuare ancora per motlo abbiamo citato soltanto le principali. Come si vede, comunque, il lavoro del programmatore può risultare notevolmente sveltito da questi strumenti.
E' quindi sempre consigliabile procurarsi qualche utility per il proprio computer. Di solito, riuntie in un certo numero, sono vendute sotto forma di programmi chiamati TOOL KIT (in inglese: scatola d'attrezzi), TOOL SET, PROGRAMMER'S AID (aiuti al programmatore) o simili, che si installano (introducono, attivano) facilmente leggendoli da dischetto, cassetta o cartuccia.
Il concetto di utility può anche essere esteso a tutti i programmi che non servono direttamente a risolcere un certo problema, ma facilitano il lavoro in generale.
Per esempio, programmi per realizzare grafici, per copiare dischetti, per verificare il corretto funzionamento del calcolatore, e così via.

DEBUG

Non sempre un programma funziona perfettamente al primo colpo; anzi, è abbastanza raro. Errori o imperfezioni nell'analisi del problema, nella stesura dell'algoritmo o nella scrittura fisica del programma causano spesso un comportamento diverso da quello voluto.
Gli errori o imperfezioni di un programma si chiamano BUG, termine inglese traducibile con pulce o insetto molesto (qualcuno lo chiama anche BACO). Il lavoro di messa a punto ed eliminazione dei problemi prende il nome di DEBUG (letteralmente "spulciare").
Certi errori sono facili da eliminare: se il programma si ferma ed appare il messaggio:

```
? SINTAX ERROR IN 120
```

vuol dire, evidentemente, che la linea 120 contiene un errore di sintassi che la rende incomprensibile per il computer.
La situazione è più complessa quando ci sono istruzioni condizionali.
Ad esempio, consideriamo queste linee di programma:

```
50 IF (A=B-2) OR (B>C-1) GOTO 70
60 A=A-1
70 ...
```

Quando le cose non vanno nel modo voluto, è difficile stabilire se l'istruzione alla linea 60 è stata eseguita oppure no. Occorre allora fare uso di qualche TECNICA DI DEBUG.
Vediamo le più comuni.

1 - Inserire delle PRINT per verificare se il programma passa da (esegue o meno) una certa linea. Nel nostro esempio, possiamo aggiungere una linea:

```
61 PRINT 61
```

Facendo GIRARE (RUN) il programma, sapremo se la linea 60 viene eseguita oppure no. Se viene eseguita, il programma prosegue con la 61 che abbiamo inserito e stampa, appunto, il numero 61 (si può far stampare qualunque cosa, ad esempio il valore di una variabile).
Se, invece, il programma salta dalla linea 50 alla 70, non esegue nè la 60 nè la 61, dunque non stampa nulla.

2 - Inserire un BREAKPOINT (punto di arresto) ed osservare i valori delle variabili. Nell'esempio, si può scrivere:

```
61 STOP
```

Quando (se) il programma esegue la linea 61, l'esecuzione si arresta con il messaggio:

```
BREAK IN 61
```

A questo punto, scriviamo direttamente:

```
PRINT A, B, C
```

e vediamo i valori delle tre variabili, verificando se sono corretti.

3 - Per seguire l'esecuzione del programma, alcuni computer possiedono il comando TRACE,od altri simili; in altre macchine è possibile aggiungerlo con un'apposita utility letta da una memoria di massa.
Il comando TRACE (traccia) di solito racchiuso tra due simboli speciali visualizza il numero di linea di tutte le istruzioni eseguite.
E' quindi possibile vedere quali istruzioni vengono eseguite e quali no, ed in quale ordine.

In ogni caso, il miglior metodo per eliminare gli errori più fastidiosi cosiste nell'evitarli fin dall'inizio, spendendo un poco di tempo in più nell'analisi del problema e nella definizione dell'algoritmo.
Questo tempo si ripaga abbondantemente nella stesura del programma, che deve soltanto tradurre in BASIC l'algoritmo, e soprattutto nella fase di debug, dove si incontrano solo problemi banali e facilmente risolvibili.
Molto importante è anche una completa DOCUMENTAZIONE del programma, cioè la spiegazione del suo funzionamento, sia su carta sia nel programma stesso (per mezzo delle istruzioni REM).
Verso la fine del corso vedremo per esteso una corretta e moderna tecnica di documentazione dei programmi BASIC.

(continua)

COMPUTER & PROGRAMMAZIONE (Parte IIIª)

ALGORITMI E DIAGRAMMI DI FLUSSO

Un algoritmo è una sequenza di operazioni che devono essere svolte per arrivare alla soluzione ottimale di un problema. Un esempio chiarirà meglio le idee:

Algoritmo per la somma di due numeri

1) Incolonnare a destra i due addendi
2) Considerare le cifre della colonna piu' a destra
3) Assumere inizialmente il riporto uguale a 0
4) Sommare le cifre della colonna in considerazione ed aggiungere il riporto
5) Se il risultato ottenuto eseguendo l' istruzione 4) è minore di 10 scrivere la cifra risultante nella colonna in considerazione, porre il riporto uguale a 0 e andare all' istruzione 7)
6) Se il risultato ottenuto eseguendo l' istruzione 4) è maggiore o uguale a 10 sottrarre 10. Scrivere la cifra risultante nella colonna in considerazione, porre il riporto uguale a 1 e andare all'istruzione 7).
7) Spostarsi sulla cifra contenuta nella colonna immediatamente a sinistra.
8) andare all'istruzione 4).

Come abbiamo potuto notare, l'algoritmo è stato scritto con una forma ordinata e precisa di esposizione che ci mette al riparo da inevitabili ambiguità come potrebbero essere quelle derivanti da un Normale Linguaggio, nel nostro caso, l'italiano.
Descrivere comunque gli algoritmi in NL comporta indubbiamente un vantaggio poiché esso è patrimonio comune di un vasto numero di persone, ma ha lo svantaggio di essere generalmente:

A) poco preciso
B) complesso
C) a volte inadeguato per certe problematiche.

Dobbiamo comunque ammettere che durante le fasi preliminari di analisi (primordiale studio) di grossi problemi il NL è tuttora usatissimo. E' risaputo che un grave errore dei programmatori alle prime armi, è quello di essere così ansiosi di scrivere il programma da non preoccuparsi di stendere prima una dettagliata specifica del programma stesso.
Questo è un approccio claudicante. Il metodo più corretto per scrivere un programma comprende tre fasi essenziali:

A) Il problema deve essere capito (questo comporta la conoscenza della natura delle strutture dei dati in ingresso).
B) Il programma deve essere predefinito con un algoritmo che ne delinei i punti principali.
C) L'algoritmo deve essere trasformato in un diagramma di flusso (per una migliore e schematica comprensione dei punti da sviluppare), per poi essere tradotto (codificato) nel linguaggio di programmazione specifico (es. Fortran Pascal, basic, C, ecc).

L'omissione di uno di questi passi può portare a gravi errori.
Un programma che comporti un piccolo numero di istruzioni, non richiede un diagramma di flusso perché la mente umana è in grado di memorizzare e sviluppare un certo numero di dati.
Quando però l'uomo deve sviluppare un programma con numerose istruzioni, allora è necessario aiuto.
In questi casi si ricorre al diagramma di flusso.
Il Diagramma di Flusso è una rappresentazione bidimensionale in forma di programma dell'intero problema che deve essere risolto dal calcolatore. Siccome i calcolatori possono svolgere solamente quattro operazioni elementari, il diagramma di flusso le deve rispettare.

Operazione di ingresso-uscita

Operazione di confronto e decisione

Procedura da sviluppare

La freccia indica la direzione di flusso dell' algoritmo.

Inizio/fine (start/stop) del DdF

Un semplice esempio chiarirà le idee sull'utilizzo dei DdF.
Immaginiamo che un meccanismo debba segnalare le macchine che passano in un cancello mediante un dispositivo a fotocellula.
L'algoritmo in NL sarebbe:

1) Poni il contatore del meccanismo uguale a zero.

2) Un'automobile è passata dalla fotocellula ? Se NO ripeti l'istruzione numero 2

3) Se SI allora incrementa il contatore di uno e vai all'istruzione numero due.

Tradotto in DdF diventerebbe:

START

Contatore = 0

macchina passata ?

contatore=contatore=1

COMPUTER & PROGRAMMAZIONE (Parte III^a)

Con questo esempio abbiamo concluso l'introduzione dei diagrammi di flusso, vediamo ora le costanti e le variabili.

COSTANTI, VARIABILI, ESPRESSIONI E LORO VALUTAZIONE

Come si sa esistono vari tipi di dati, come ad esempio i numeri, le stringhe di caratteri, ecc ecc. In questo paragrafo faremo riferimento a numeri interi grandi a piacere e tutte le considerazioni che seguono possono essere generalizzate a qualsiasi tipo di dato.
Il tipo di dato "intero" è stato scelto sia perché è già sicuramente a conoscenza di chi legge, sia per la ricchezza delle sue operazioni. Iniziamo con l'osservare la seguente scrittura :

(5x-y)(x+3)

Tutti sanno riconoscere l'espressione numerica dove x e y sono due VARIABILI e i numeri 5 e 3 sono dette COSTANTI. Sappiamo anche che l'espressione non ha un valore particolare fino a che non si attribuiscono dei valori specifici alle variabili come ad esempio se :

x = 1 e y = 3 in (5x-y)(x+3)

possiamo dire dopo un processo di valutazione e di calcolo che l'espressione vale 8.
Questi concetti di costante, variabile, espressione e valutazione di una espressione costituiscono degli ingredienti essenziali della programmazione e verranno ripresi ed estesi più volte nelle nostre lezioni.
D'ora in avanti supporremo che l'esecutore disponga di un dispositivo particolare chiamato VALUTATORE che riceve una espressione aritmetica intera e, dopo un processo di valutazione,restituisce il risultato.
Per fare ciò egli deve poter prelevare i valori associati alle variabili che compaiono nell'espressione come ci si può rendere conto dall'esempio appena fatto.
Chiameremo AMBIENTE un insieme di variabili con i valori loro associati.
La valutazione di un'espressione richiede quindi la presenza di un ambiente in cui valutarla come :

(5x-y)(x+3)------->VALUTATORE-------->Valore della espressione

^
!
!
!
AMBIENTE DI VALUTAZIONE
x = 1   y = 3

L'OPERAZIONE DI ASSEGNAMENTO

Come già sappiamo, una variabile è un nome a cui è associato un valore che viene reperito nel processo di valutazione delle espressioni.
Nel suddetto esempio abbiamo implicitamente fatto riferimento a questa proprietà in istruzioni del tipo:

assegna a x il valore 1

Non a caso questa istruzione viene detta assegnamento di una variabile : essa ha l'effetto di costruire un'associazione tra x ed il nuovo valore specificato cancellando ogni eventuale associazione precedente.
La simbologia lega l'operazione di assegnamento con il segno matematico di uguaglianza (=) oppure, in alternativa, il simbolo --›.

x ‹-- 1 equivale a dire x = 1

L'espressione a sinistra del simbolo

"‹--"

di assegnamento viene valutata ed assegnata alla variabile di sinistra del simbolo

"‹-"

nel senso appena descritto.
Più semplicemente possiamo dire che nel caso

x = 1 (o x ‹-- 1)

il vecchio valore di x viene sostituito da quello nuovo.
Per intenderci, possiamo pensare ad una variabile come una scatola etichettata che può contenere un unico valore che può essere sostituito dopo una successiva operazione di assegnamento.
Per questo motivo si parla di "contenuto" di una variabile invece di "valore" di una variabile anche se si possono usare scambievolmente le due espressioni.
La lettura di una variabile, cioè il processo di valutazione di una variabile non è distruttiva; la variabile, in questo caso x, continua ad essere legata allo stesso valore 1.
La scrittura di una variabile, cioè l'assegnamento di un valore alla variabile, è invece distruttivo : il suo valore non può essere reperito in quanto sostituito dal nuovo valore assegnato.

(continua)

DENTRO L'MSX (Parte Terza)

IL BASIC MSX

Il BASIC MSX è sostanzialmente il BASIC Microsoft standard versione 4.5 con numerose estensioni nell'area della grafica e della produzione di suoni.
Le principali mancanze sono invece le Procedures, le strutture WHILE/WEND e REPEAT/UNTIL e la completa abbreviazione delle parole chiave.
Viene reso disponibile un editor a schermo intero che permette di correggere una qualsiasi linea dello schermo. La correzione effettiva della linea avviene quando viene battuto il tasto [RETURN] mentre il cursore si trova su qualsiasi carattere della linea in questione. Nella fase di introduzione non viene effettuato alcun controllo di correttezza.
Una linea può essere composta al massimo da 255 caratteri e può contenere più di una istruzione.
Più istruzioni che si trovino sula stessa linea devono essere separate dai due punti. Un'istruzione REM fa sì che l'esecuzione prosegua dalla linea successiva. I numeri di linea devono essere compresi nell'intervallo 0-65529, compresi gli estremi.
Gli spazi non sono necessari e vengono ignorati dall'interprete (tranne quando fanno parte di una stringa). Una conseguenza di questo fatto è che le parole chiave non possono essere parte di nomi di variabili (dei quali sono riconosciuti solo i primi due caratteri).
Vengono considerate diverse le lettere maiuscole e quelle minuscole. Le parole chiave possono essere scritte sia maiuscole che minuscole.
A fianco della tastiera principale vi sono due gruppi di tasti. Sulla destra il gruppo dei tasti per l'editing - i tasti con le frecce che indicano la direzione dello spostamenteo del cursore e le opzioni di inserimento e cancellazione.
Sulla sinistra c'è un insieme di cinque tasti funzionali. Al momento dell'accensione questi tasti sono predefiniti con dieci funzioni.
Quelle corrispondenti ai tasti funzione "minuscoli" (cioè senza che sia premuto il tasto [SHIFT]) sono mostrate sull'ultima linea dello schermo. Questa visualizzazione può essere evitata, pur mantenedo la definizione delle funzioni, tramite il comando KEY OFF.
Per ridefinire uno qualsiasi dei tasti funzione, si usa l'istruzione KEY x,"stringa".
Per esempio, si può definire nel seguente modo il tasto funzione 1 in modo che serva per mandare in esecuzione un programma.

```
KEY 1,"RUN"+CHR$(13)
```

La visualizzazione di una lista di programma o un'esecuzione può essere interrotta premendo il tasto [STOP] una volta. Premendolo nuovamente viene riattivata l'azione precedentemente interrotta.

Per terminare definitivamente l'operazione in corso si devono premere contemporaneamente lo [STOP] e il tasto [CTRL] (CONTROL).
Per coloro che non hanno familiarità con il BASIC Microsoft è opportuno porre in evidenza una serie di caratteristiche:

1 – La messa a punto dei programmi è semplificata dall'uso delle istruzioni TRON e TROFF. Se viene eseguito il comando TRON, direttamente o all'interno di un programma, questo fa sì che venga visualizzato il numero di ogni linea che viene eseguita; solitamente eventuali altre visualizzazioni sono sovrascritte.

2 – Blocchi di linee possono essere cancellati con l'istruzione DELETE X-Y, mentre le linee di un programma possono essere rinumerate tramite il comando RENUM X,Y.
Questa possibilità si rivela utile nell'identificare eventuali salti a numeri di linea non esistenti. In tal caso infatti il BASIC MSX non segnala un errore; viene invece eseguita la linea con numero immediatamente superiore a quello indicato nel salto.

3 – Tutte le variabili che iniziano con un particolare carattere, o con un insieme di caratteri, possono essere dichiarate di un particolare tipo: DEFINT per variabili intere, DEFSNG per variabili in singola precisione, DEFDBL in doppia precisione e DEGSTR per le stringhe.

4 – I vettori pososno essere cancellati per mezzo del comando ERASE.

5 – Al momento dell'accensione lo spazio allocato per la memorizzazione delle stringhe è di 200 byte. Un'ulteriore allocazione necessita dell'uso di un comando di CLEAR. Ad esempio, per riservare 1000 byte si usa CLEAR 1000.

6 – La funzione FRE(0) ritorna l'ammontare della memoria disponibile per la memorizzazione del programma.

7 – La funzione per le stringhe MID$ può essere usata per sostituire uno o più caratteri in un'espressione.

8 – La struttura GOSUB/RETURN consente di specificare un indirizzo di ritorno: RETURN "numero di linea".

DENTRO L'MSX (Parte Terza)

VARIABILI E FUNZIONI

I nomi delle variabili possono essere di qualsiasi lunghezza, ma devono iniziare con una lettera. Solo i primi due caratteri sono significativi. Le lettere minuscole vengono distinte da quelle maiuscole.
Non è necessario inizializzare le variabili: viene assunto il valore zero. Le variabili di tipo vettore non devono necessariamente essere dimensionate, tranne quando il numero di elementi supera undici: quindi S(0) e S(10), per esempio, non hanno bisogno di un esplicito dimensionamento.
Una variabile che non è dichiarata di un tipo particolare si assume che sia reale in doppia precisione, e viene memorizzata con quattordici cifre significative.
Le variabili dichiarate in singola precisione sono memorizzate con una massima precisione di sei cifre.
Gli interi sono compresi tra -32768 e 32767.
Una variabile in singola precisione viene distinta tramite un punto esclamativo alla fine, come SP!, mentre un'intera viene contraddistinta da un segno di percentuale (%). Se è necessario dichiarare una variabile in doppia precisione, viene usato il simbolo #.
Come accennato in precedenza, le varianti dell'istruzione

DEF

possono essere utilizzate per definire globalmente i tipi delle variabili.
Se ad esempio viene usata l'istruzione

DEFINT A

allora tutte le variabili che iniziano con A verranno trattate come variabili intere.
Inoltre non verrà fatta alcuna distinzione tra 'A' e 'A%'.
Il dimensionamento delle variabili è molto semplice: si consideri a questo proposito l'istruzione

DIM X(20,40),Y(80,160)

che definisce due matrici rettangolari; la prima ha 20 righe e 40 colonne mentre la seconda ne ha 80 per 160.
Esiste il limite di 255 sul numero delle dimensioni, ma il numero degli elementi è limitato solamente dalla disponibilità della memoria.
Poiché una qualsiasi variabile vettore che abbia meno di 11 elementi non deve essere esplicitamente dimensionata, ne consegue che, per un utilizzo ottimale dello spazio di memoria, è opportuno dimensionare quelle che hanno un numero di elementi minore di 11.
Il BASIC MSX comprende un'istruzione

SWAP X,Y

che serve per scambiare il contenuto delle variabili X e Y. Le variabili in questione devono però essere del medesimo tipo. Un'altra istruzione in relazione con le variabili è la

VARPTR X

che ritrona la locazione di memoria dell'oggetto specificato - se quest'ultimo è stato dichiarato.

Le variabili di sistema dell'MSX sono:

TIME – E', in un certo senso, l'orologio del sistema. Viene incrementato ogni 1/50 di secondo.

BASE(N) – Ritorna la locazione della tabella specificata della RAM video, indipendentemente dalla modalità di visualizzazione.

VDP(N) – Ritorna il valore del registro a sola scrittura specificato dal VDP, o viceversa il registro di stato di quest'ultimo.

SPRITE$(PATTERN#) – Questa variabile viene usata per definire ognuna delle 256 possibili 8*8 forme di sprite, oppure una delle 64 forme 16*16. Contiene una stringa di 32 caratteri. Il codice di ciascuno di questi caratteri determina la configurazione dei bit in un byte della definizione dello sprite.

FUNZIONI

Sono disponibili tute le funzioni che siamo abituati a trovare nel BASIC.
In particolare, se X è il nome di una arbitraria:

ASC(X$), CHR$(X), LEN(X$), INT(X), EXP(X), ABS(X), LOG(X), HEX$(X), OCT$(X), BIN$(X), SGN(X), RND(X), MID$(X$,N,n), RIGHT$(X$,N), LEFT$(X$,N), STR$(X), VAL(X$), STRING$(N,X$), INSTR(X$,Y$), ATN(X), COS(X), SIN(X), SQR(X), TAN(X), TAB(n), SPC(n), INKEY$, INPUT$(n), PEEK(X), POKE X,N, FRE(0), FRE"", USR X(n), VARPTR(X), VARPTR(#X), CINT(X), POS(n) e LPOS(n).

Le estensioni comprendono:

CDBL(X) e CSGN(X)
che convertono X rispettivamente a doppia e singola precisione.

DENTRO L'MSX (Parte Terza)

VPEEK(n) e VPOKE n,X
sono le istruzioni equivalenti, per la RAM video, di PEEK e POKE.

STICK(n) e STRIG(n)
ritornano la direzione e lo stato di trigger del joystick (oppure del cursore e della barra di spaziatura).

POINT(x,y)
ritorna il colore del pixel che si trova alle coordinate X e Y.

PLAY(canale)
indica lo stato di una o di tutte le code musicali.

EOF(file#)
ritorna -1 se è stata raggiunta la fine di un file sequenziale, alltrimenti 0.

PAD(n):
in questo caso n determina quale parametro dello statro di un paddel a controllo tattile viene ritornato.

PDL(paddel#)
ritorna il valore di un paddle.

ISTRUZIONI PER LA GRAFICA

L'intera gamma dei 15 colori più il trasparente è disponibile in tutte le modalità di visualizzazione.
Le istruzioni per la grafica possono essere divise in tre categorie:

1 - Formattazione generale e colore
2 - Sprite
3 - Alta risoluzione

COMANDI DI USO GENERALE

La modalità di visualizzazione sullo schermo viene impostata con l'istruzione SCREEN:

SCREEN 0 modalità testo 40*24
SCREEN 1 modalità testo 32*24
SCREEN 2 modalità grafica ad alta risoluzione (HRG)
SCREEN 3 modalità multicolore

Questi comandi possono essere utilizzati anche per impostare la dimensione degli sprite, la velocità di trasferimento dati su cassetta e le opzioni per la stampante.
Quando vengono usate per selezionare una modalità testo, l'insieme dei caratteri viene copiato dalla ROM alla VRAM. L'istruzione SCREEN non deve essere usata qualora l'utente abbia la necessità di fare riferimento ad un qualsiasi carattere ridefinito.

MODALITA' TESTO

La larghezza effettiva dello schermo sia per la modalità a 40 colonne sia per quella a 32, può essere diminuita con l'istruzione

WIDTH

(ad esempio WIDTH 20 per avere lo schermo su 20 colonne soltanto).
Si noti che in entrambe le modalità testo la colonna più a destra e quella più a sinistra non vengono usate.
La maggior parte dei ricevitori TV non visualizzano la colonna più a sinistra.
In entrambe le modalità i caratteri possono essere messi in una colonna particolare scrivendo sulla tabella dei nomi della RAM video. Ad esempio, per mettere una A maiuscola nella prima riga delle due colonne più a sinistra nella modalità a 40 colonne, si usa la

VPOKE

delle locazioni 0 e 1 con il valore 65.
I colori globali per una qualsiasi modalità di visualizzazione vengono impostati tramite l'istruzione

COLOR.

Il formato della sua sintassi è il seguente:

COLOR primo-piano, sfondo, bordo

L'impostazione di default è 15,4,4. Nel modo a 40 colonne il colore del bordo non può essere specificato separatamente e assume quello dello sfondo. Nei modi schermo 1, 2 e 3 si possono ottenere uno sfondo ed un bordo color grigio inchiostro con l'istruzione COLOR 14,1,1.
Si noti che nelle modalità grafiche il nuovo colore dello sfondo diventa effettivo solo dopo un'istruzione CLS.
Nella modalità testo a 40 colonne, il VDP non consente di usare altro colore.
Non è possibile usare gli sprite. Viceversa nel modo a 32 colonne il VDP visualizza il testo in un colore differente da quelli scelti per mezzo dell'istruzione COLOR. Però a questo scopo non esiste comando BASIC. Quindi per ottenere sullo schermo più di un colore di primo piano o di sfondo, è necessario apportare dei cambiamenti alla tabella dei colori della VRAM. Per far ciò si usa una VPOKE.
Un'operazione del tutto simile deve essere effettuata per ridefinire dei caratteri.

(Continua)

SPECIALE SIMULAZIONE

Fin da quando comparvero i primi Home Computer i simulatori di volo sono sempre stati uno dei generi preferiti e maggiormente sfruttati dalle case produttrici di programmi.
Indipendentemente dal fatto che li apprezziate o meno (anche se è un dato di fatto che la maggior parte non ci gioca per l'eccessivo impegno necessario ad imparare ad usare i numerosi comandi che nella maggior parte dei casi sono tutt'altro che agevoli) rappresentano indubbiamente uno degli sviluppi più interessanti intrapreso dal software applicativo.
Con l'avvento dei 16-Bit poi si è finalmente aggiunto, grazie alle ampliatissime possibilità grafiche, quello straordinario

realismo che fa di alcuni programmi dei veri e propri capolavori.
Partendo dal fatto che un simulatore di volo è prima di tutto, per la stessa definizione, un simultaore, risulta fin da subito evidente come il suo obiettivo principale sia quello di riprodurre quanto più fedelmente possibile la realtà in tutti i suoi più svariati aspetti.
Tornando agli albori del software, quegli ormai dimenticati titoli che allora erano tanto acclamati non potrebbero non farci sorridere nella loro rudimentale essenzialità; anche se nonostante tutto, con un pò di fantasia, si poteva ugualmente calarsi nei panni dell'intrepido pilota impegnato in evoluzioni mozzafiato.
Ad ogni modo uno dei problemi principali per quei così detti simulatori della prima generazione (grafica a parte) era una totale mancanza nella maggior parte di un vero e proprio obiettivo conforme ad una specifica missione.
Si finiva così per vagabondare per il cielo cercando di schiantarsi il più tardi possibile.
Una svolta determinate venne probabilmente con il celeberrimo

GUNSHIP

della Microprose (anche se altri programmi come

SOLO FLIGHT e SUPER HUEY

avevano già imboccato la strada buona) che accompagnava una più dignitosa simulazione di elicottero con una complessa quanto altamente coinvolgente struttura di supporto che prevedeva la scelta degli armamenti, delle missioni, la possibilità di fare carriera con il proprio pilota, aspetto che fu forse la ciliegina sulla torta e che permise al programma di conseguire onorificienze e riconoscimenti un pò da tutta la stampa specializzata.
La svolta successiva che ci ha permesso di arrivare a quelli della terza generazione è stato indubbiamente l'avvento dei computer a 16-bit che grazie alle loro caratteristiche tecniche di gran lunga superiori hanno consentito di ottenere un realismo che fino a qualche anno fa poteva sembrare irrangiungibile.
Il primo programma che sconvolse tutti gli utenti con una solida e oltremodo fluida grafica vettoriale in tre dimensioni e con la possibilità di svariare il punto di vista dell'utente fu l'ultrafamoso

F/A-18 INTERCEPTOR

dell'Electronic Arts.
Per la prima volta il panorama non era più limitato quasi solamente a due zone di diverso colore rappresentanti presumibilmente il cielo e la terra e alcune indecifrabili figure triangolari, ma bensì un'abbondanza di dettagli e piccoli tocchi che aumentavano notevolmente quella strana sensazione "d'esserci veramente".
In ogni caso INTERCEPTOR fu anche criticato per la sua irrealisticità nei movimenti, anche se noi crediamo che questo abbia facilitato una così improvvisa diffusione aprendo la strada a tutti i seguenti programmi sempre più evoluti come

F15 STRIKE EAGLE II della Microprose,

FIGHTER BOMBER dell'Activision,

F29 RETALIATOR dell'Ocean,

LHX ATTACK CHOPPER dell'Electronics Arts,

F16 FALCON della Spectrum Holobyte,

F16 COMBAT PILOT della Digital Integration

e via dicendo...
Ognuno di questi simulatori della nuova generazione è caratterizzato da un numero sempre maggiore di opzioni che esulano da quella che è la fase di volo vera e propria.
Il "background" del gioco si è fatto via via più esteso consentendo al nuovo cadetto di vivere un'esperienza quanto più completa e attinente alla realtà.
E così ecco comparire uffici per l'arruolamento, accuratissime sezioni d'allenamento, varietà di missioni da affrontare, possibilità di determinare autonomamente l'armamento da adottare e soprattutto (come già succedeva in GUNSHIP) doversi preoccupare del proprio "curriculum interattivo" che si aggiorna progressivamente procedendo nelle missioni.
Aspetto questo che è stato poi ulteriormente implementato anche per quanto riguarda lo scenario di ogni missione che subisce le conseguenze delle vostre azioni precedenti e così se ad esempio avrete già distrutto tutte le postazioni antiaereo (SAM) la volta dopo avrete un problema in meno a cui fare attenzione, per non parlare poi del campo di battaglia di F29 che va avanti da solo autonomamente lasciandovi l'iniziativa di intervenire a vostra discrezione nel caso non si tratti di un bersaglio primario.
Verrebbe a questo punto da porsi una domanda, ma che cosa caratterizza un simulatore dell'ultima generazione?
Probabilmente fino ad oggi il più completo e tecnicamente accurato deve essere considerato

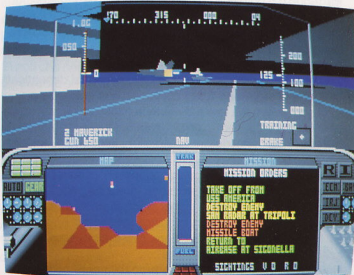
FLIGHT SIMULATOR 4 (FS4)

della Sublogic/Microsoft che non solo presenta una fedele e oltremodo veritiera simulazione di un Cessna 182, ma anche degli scenari dinamici con tanto di strade trafficate e minuziosissimi particolari come ad esempio degli yachts per aggiungere un ulteriore tocco di realismo.
E in più c'è perfino un costruction-set che permette di modificare molti dei parametri pre-impostati.
Analizziamo comunque le caratteristiche comuni di tutti i simulatori citati in precedenza per vedere dove questi ancora pecchino di verismo.
Dunque ognuno vi mette al comando di un aereo, oppure una lista fra cui scegliere ognuno dei quali viene presentato con le sue specifiche caratteristiche relative alle dimensioni, velocità, autonomia, punto di stallo, resistenza alla forza gravitazionale, inclinazione massima di salita etc..
Ci sono poi i vari scenari e le numerose missioni con i loro particolari obiettivi da portare a termine vedendosela direttamente con le forze nemiche che vi impegneranno in uno scontro diretto, il problema dei rifornimenti e della strategia vincente da adottare.
Ma tutte queste caratteristiche, secondo il parere di un vero pilota militare, per quanto sicuramente indispensabili, non sono sufficienti per assicurare quel tanto agognato realismo che ogni simulazione si prepone come meta finale.
Quello che caratterizza infatti una vera missione odierna e che può influenzarne l'esito anche al di là dell'abilità dei piloti o delle caratteristiche tecniche dei velivoli è infatti l'affiatamento che si riesce a formare all'interno di una pattuglia che determina la sincronia con la quale vengono messe in atto le varie manovre fondamentali per la riuscita dell'azione.
Se ora andiamo a rivedere tutti i simulatori sopracitati possiamo facilmente verificare come la maggior parte di questi proponga un solo aereo in missione contro moltitudini di agguerritissimi nemici, cosa che rende piuttosto evidente una componente irrealistica ancora lungi dall'essere completamente annullata.
Fortunatamente però ci si sta già muovendo anche in questa direzione come testimoniano alcuni programmi in circolazione che cercano di ovviare a questo inconveniente in maniera piuttosto efficace.
Ci riferiamo in particolare sia ai due simulatori della Lucasfilm

BATTLEHAWKS/THEIR FINEST HOUR

sia ad

A10 TANK KILLER

della Dynamix che si possono considerare per gli accorgimenti che vado ad esporvi ancora un passo in avanti rispetto agli altri.

Entrambi i simulatori consentono o necessitano di essere affrontati tenendo bene in considerazione per la riuscita della missione che si sta effettuando un'operazione di gruppo con più unità partecipanti.
In

BATTLEHAWKS

ad esempio vi sarà certo capitato di sentirvi ormai perduti con un aereo nemico proprio dietro di voi pronto a colpirvi quando è intervenuto provvidenzialmente un vostro compagno a salvarvi da un'ormai disperata situazione.
In

A10

invece viene adottato l'interessantissimo collegamento via radio che vi informa prontamente se state andando fuori posizione o se un vostro compagno è sotto il fuoco nemico incaricandovi di andare a dargli una mano o magari di dividervi per un'azione differenziata.
Ma oltre a questo

THEIR FINEST HOUR

presenta anche un nuovo accorgimento che lo caratterizza ulteriormente e che crediamo diverrà un punto fisso per i prossimi simulatori in fase di sviluppo, mi riferisco all'intelligenza simulata dei piloti da voi non direttamente controllati che evolvono individualmente le loro caratteristiche personali permettendovi di trasferirli a vostro piacimento da una base ad un'altra nel tentativo di allestire una squadriglia quanto più idonea possibile per affrontare una determinata missione.

QUALE COMPUTER?

Quale computer potrebbe rappresentare la miglior base per l'utilizzo di un simulatore di volo?
Senza dubbio la scelta più indicata case sull'MS-DOS modello AT (IBM o qualsiasi PC compatibile) piuttosto che sul più lento modello XT, che si rivelerebbe poco adatto ad elaborare la grafica e i dati di un flight simulator qualsiasi.
Tanto per cominciare, molte delle migliori simulazioni di volo vengono sviluppate per PC MS-DOS compatibili.
Inoltre, la versione per PC di Flight Simulator della SubLogic è finora l'unica disponiblie aggiornata: ha infatti raggiunto la quarta versione mentre quelle per altri personal non hanno superato la seconda, addirittura la prima per l'MSX.

Molti simulatori di volo sfruttano bene le capacità grafiche dell'ottima scheda VGA del PC (256 colori su schermo contemporaneamente) così come l'opzione per il joystick analogico e altri accessori simili per consentire movimenti più precisi se comparati a quelli prodotti dagli input digitali degli usuali joystick.
Così come i PC più veloci (286, 386 e ora 486) producono animazioni molto più fluide nel volo, si è pensato di sfruttare l'hardware specifico che costituisce oggi la parte più potente dei nuovi PC.
FLIGHT SIMULATOR 4 usa le istruzioni specifiche del 286 e del 386 e FALCON 3.0 della Spectrum Holobyte sarà in grado di sfruttare il coprocessore matematico se presente.
E la miglioria apportata al secondo simulatore citato gli dona un incredibile realismo aereodinamico.
Il sonoro sta subendo anch'esso un certo sviluppo grazie alla sempre maggiore diffusione di schede sonore.
La maggior parte degli sviluppatori utilizza l'ADLIB mentre altre schede - GAMES BLASTER, ROLAND LAPC-1 - stanno assumendo sempre maggiore importanza.
E proseguendo coi vantaggi, c'è da dire che sono stati creati una serie di 'optional' progettati tenendo presente il PC come piattaforma Hardware, ma possono essere facilmente adattati per l'uso con altre macchine.
Il MAXX è un joystick specifico 'tipo cloche' molto adatto per le simulazioni di grossi velivoli o di aereoplani civili.
La ADVANCED GRAVIS offre un joystick molto più tradizionale che dà una maggiore impressione di ambiente da combattimento aereo.
Tuttavia, la SubLogic possiede la serie definitiva di controlli di volo - almeno per ora.
Chiamato FLIGHT CONTROLS 1, il marchingegno comprende una cloche, acceleratore ed altri interruttori come un selettore di flap il tutto collegato ad un ampio pannello di controllo. E' disponibile persino un set di pedali di controllo.

Il punto debole del PC?
Uno solo, il prezzo.

Un PC 386 dotato di scheda VGA, monitor a colori e hard disk abbastanza accessibile costa comunque più di tre milioni, vedi l'ultimissimo Amstrad.
Sebbene i modelli 286 con VGA stanno diventando più accessibili stiamo intorno al milione e mezzo come minimo - non è ancora una fascia di prezzo adatta alla maggior parte degli appassionati di simulazione.
A questo punto, dobbiamo solo sperare che qualche casa di buona volontà produttrice di hardware produca un PC AT con VGA, presa joystick di serie a scheda sonora che non superi - nel suo prezzo globale - il milione.

Continuiamo a sperare...

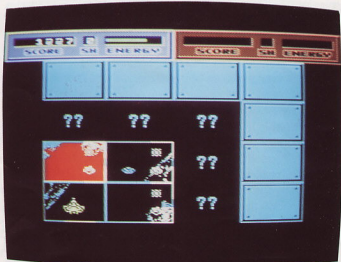
3D POOL

Prodotto da FIREBIRD
Potete trovarlo da:
Msx Club Italia - Cp 34 - 20075 Lodi Centro (MI)
Disponibile sia su disco che su cassetta
Prezzo: 7.000 lire

Come si poteva restituire la giovinezza a un tipo di gioco già ben sviluppato su microcomputer?
I biliardi automatici avevano raggiunto il top della qualità e della ricchezza.
La novità di questa edizione è lo sviluppo del gioco in 3 Dimensioni, cosicché gli ideatori di soft rilanciano la moda del "pool".
Dopo il superbo Billiard Simulator su ST, 3D Pool vi offre uno scenario e un gioco molto semplici, ma comunque soddisfacenti per gli appassionati del genere.
Unico punto interessante della partita, il giocatore non deve soltanto orientare la stecca, ma anche girare introno al tavolo per scegliere il suo angolo di tiro.
Maneggiato con il joystick, l'orientamento 3D della scena è molto fluido.
Ma, per contro, tutti gli elementi del gioco sono semplificati all'estremo, il che non accadeva per Billiard Simulator.
La grafica non è eccellente e i menù di selezione si riducono a una fredda pagina di testo, ma bisogna considerare che dovendo girare anche sugli Msx 1 il lavoro fatto dai programmatori è stato notevole.
Niente fronzoli dunque, solo un gioco efficace da vedere.
Riservato agli appassionati.
Sono previsti adattamenti su tutti i microcomputer.

BLASTEROIDS

Prodotto da IMAGE WORKS
Lo trovate da:
Msx Club Italia - Cp 34 - 20075 Lodi Centro (MI)
Disponibile solo su disco
Prezzo: 7.000 lire

E' dentro alla vecchia terracotta che vengono preparate le migliori zuppe.
Vi chiederete, è forse ora di pranzo che si parla di pentole e zupper?
No! Non è che un modo un po' insolito per riparlarvi, diciamo così, di una vecchia "zuppa".

Riprendendo il tema di Asteroid (il precursore dei giochi d'azione spaziale), Blasteroids ripropone in modo semplice questo tema di gioco.

Dovete esplorare diversi settori della galassia pieni di meteoriti e vascelli extraterrestri.

Motivo di interesse è dato dal fatto che la distruzione dei vascelli alieni vi permette di migliorare l'armamento e la potenza di propulsione del vostro vascello.

Il vostro ultimo obiettivo è annientare Mukor (una creatura immonda sorgente di tutti i vostri fastidi), forte dell'esperienza e delle armi acquisite durante le fasi precedenti.

Gli effetti grafici sono eccellenti, ma anche le animazioni multiple che prevedono un numero considerevole di oggetti nello schermo.

Pur avendo visto anche la versione Amiga, dobbiamo dire che su MSX (funziona anche con l'Msx 1) la grafica è buona anche se non al livello di giochi per l'Msx 2 o la stessa versione Amiga.
Il gioco funziona sotto MSX-DOS e quindi esiste solo su disco.

ARKANOID

Prodotto da TAITO
Lo trovate da:
Msx Club Italia - Cp 34 - 20075 Lodi Centro (MI)
Disponibile sia su disco che su cassetta
Prezzo: 7000 lire

Lo scenario di Arkanoid è assai stupido e privo di interesse. Esso si articola, in breve, nella storia di un vascello spaziale attaccato nello spazio dai soliti misteriosi alieni, da cui si distacca una capsula di salvataggio che va a finire in un complesso Dho di energia pura, e così via.
La sola cosa da mettere in rilievo in questo game (a parte il fatto che resta sempre un buonissimo test per i vostri riflessi), è che questa è la versione ufficiale Msx della versione arcade e che la conversione è decisamente ottica, addirittura migliore della versione per il Pc Ibm.
L'animazione è impeccabile, e la gestione del sonoro superba.
Partendo da uno spaccamattoni che ricorda il primo mitico Breakout, la Taito è riuscita a farne un Hit dove i differenti bonus a disposizione del giocatore rendono difficile ma non irraggiungibile l'obiettivo finale.

POSTA

Spett. redazione di C16/MSX, sono un affezionato lettore della vostra grandiosa rivista che compero ogni mese in edicola.
Posseggo un C16 da due anni, ma ora vorrei passare ad un home-computer più potente. Da qui la mia decisione di passare dal C16 all'MSX.
E' una scelta ragionevole?
Potreste darmi delle informazioni tecniche su un depliant che abbia una foto dell'MSX?
Ho trovato sul catalogo Vestro un MSX compatibile (la pagina dove viene indicato è nella busta).
Mi posso fidare di un computer così?
Quali differenze ci sono tra questo e un vero MSX?
E per ultimo: i vostri giochi funzionano su questo?
Vi sarei molto grato se potreste risolvere i miei problemi. Vi faccio inoltre i complimenti per la vostra rivista.

Gian Pietro BRUSETTI - Clusone (BG)

Cominciamo dalla fine.
Sì, i nostri giochi con questo MSX.
Non c'è nessuna differenza tra questo e un "vero" MSX.
Per "fidarsi" di questo MSX ci si può fidare nel senso della qualità, ma il vero problema è che, purtroppo, i computer MSX non vengono più prodotti se non in Giappone e in versioni molto diverse da quelli che ci sono in Italia.
Come dicevamo, ormai è molto tempo che la produzione degli MSX è stata sospesa ed è praticamente impossibile trovare sia il software, sia i computer e le periferiche.
Un buon consiglio è di non comprare questa macchina: la cosa migliore è di comprare un validissimo e ormai diffusissimo Commodore AMIGA.

◆

Vi scrivo perché è la prima volta che possiedo un computer C16 è non so usarlo.
Se vorreste gentilmente mandarmi tutte le istruzioni per l'uso: ve ne sarei molto grato!
Vorrei complimentarmi per la vostra rivista che, anche se è la prima volta che la compro, è veramente magnifica e non saprei proprio come avrei fatto senza.
Spero che la vostra rivista diventi sempre più bella e aggiornata di così.
Vi saluto e vi sarò molto riconoscente se mi fate questa cortesia GRAZIE TANTE.

Remo DI ROSA - Sulmona (AQ)

Mi spiace deludere questo giovanissimo lettore così entusiasta, ma non ci è possibile accontentarlo.
Purtroppo l'unica soluzione possibile è rivolgersi al rivenditore presso cui hai comprato il tuo Commodore 16 oppure direttamente alla

COMMODORE ITALIA di Cinisello Balsamo (MI).

Altra soluzione potrebbe essere quella di rivolgersi ad una libreria specializzata, anche se difficilmente sarà possibile trovare qualcosa sul C16.

◆

Spett. redazione, sono un grande affezionato alla vostra mega rivista e vi chiedo se nel prossimo numero ci fossero i seguenti giochi: Berretti Verdi; Carrier Command; Thundercats; Yes, prime minister for you; MASK; Xybots e infine un programma per trasportare i giochi da cassetta a disco.
Per migliorare la vostra rivista vi consiglio di mettere la rubrica IL MERCATO DI HARDWARE E SOFTWARE; mettere più pagine di recensione giochi rappresentando la schermata del gioco; mettere anche delle pagine dove vengono pubblicati listati di giochi o programmi vari.
Infine vi chiedo di stampare nella copertina le schermate dei giochi spiegando in modo sintetico di che cosa si tratta.
(Continuate così che siete forti!!!)

Andrea MANUZZATO - Breganze (VI)

Prima di tutto voglio ringraziare il nostro giovane amico per gli splendidi disegni che ci hai inviato: sono davvero belli.
Per quento riguarda i consigli che ci dai, come vedi stiamo migliorando sempre più e già da questo numero pubblichiamo delle nuove recensioni e presto troverai anche il gioco Green Beret (Berretti Verdi) che ci hai chiesto.
Per quanto riguarda gli altri giochi, non tutti esistono in versione MSX: l'unico che abbiamo visto è Xybots, ma non perdere le speranze e mandaci altri consigli e disegni.

POSTA

Spett. GRUPPO..., Prego esaminare la possibilità di inviarmi il n. 16 di MSX DISK (completo di dischetto e manualetto) con pagamento contrassegno o altra forma che cortesemente vogliate indicarmi. Con la presente chiedo anche se esiste o avete già prodotto o comunque dove si possa reperire un compilatore PASCAL o TURBO PASCAL per MSX e quali prezzi sono eventualmente applicati.
Ringraziando, porgo distinti saluti e i complimenti per la vostra rivista che da poco ho scoperto.

Franco D'ANDREA - Chiavari (GE)

Sembra incredibile, ma quando in redazione arrivano copie degli ultimi numeri di MSX DISK stampati, queste vanno subito esaurite come arretrati.
Così non possiamo accontentarla, ma se le interessa può procurarsi una copia del dischetto con la fotocopia del relativo manualetto all'MSX CLUB ITALIA di Lodi ad un prezzo di 7000 lire. Anche il compilatore TURBO PASCAL le potrà essere fornito dal club di Lodi, ad un prezzo non superiore a 10-15.000 lire manuale escluso. L'indirizzo è: MSX CLUB ITALIA - Cp 34 -20075 Lodi Centro (MI).

Cara redazione di C16/MSX sono un ragazzo di 13 anni, molto affezionato a voi e devo farmi i miei più vivi complimenti per la vostra rivista che naviga a gonfie vele.
Vivo in una città di nome Romano d'Ezzelino, ma siccome ho la città di Bassano del Grappa appiccicata alla mia, vado sempre a comperare da un mio amico edicolante i vostri dischi. A Bassano ci sono parecchi negozi di computers e Videogame, ma purtroppo pochi, o che sappia io nessuno, hanno quelle "COSE" che vanno bene al mio MSX2 Philips. Come posso risolvere questo problema?
Spero che leggiate la mia lettera e che mi rispondiate al più presto. Grazie.

Alessandro MILANI - Romano d'Ezzelino (VI)

Una soluzione al tuo problema può essere, innanzitutto, quella di entrare in contatto con altri utenti Msx con cui scambiare programmi, informazioni e magari anche Hardware. In alternativa puoi rivolgerti ad un club specializzato come l'Msx Club Italia. L'indirizzo è sempre lo stesso, C.P. 34 - 20075 LODI CENTRO (MI). Ti ricordo che inviando due francobolli da 500 lire riceverai il catalogo completo del software disponibile.
Tra le altre cose voglio informare chi ha già ricevuto il catalogo che tra poco riceveranno anche l'aggiornamento con gli ultimi arrivi: 150 dischi da 720k pieni di novità!

- ENDLESS NIGHT CD 01034
- BEAUTIFUL GIRLS CD 01035
- SUNLIGHT MELODIES DGC 1020

3 COMPACT DISC
AL PREZZO DI L. 29.900

Scopri nuove magiche atmosfere con le più belle melodie d'amore

SERIE "STRUMENTALI"

Desidero ricevere l'offerta "MUSIC FOR LOVERS 2" **cod. CD9**
Allego assegno ☐ ricevuta versamento ☐
+ L. 2.500 quale contributo spese postali

NOME COGNOME

VIA N.

C.A.P. CITTÀ

Firma

Compilare il coupon allegando ricevuta (o fotocopia) del versamento effettuato sul C/C n. 11319209 intestato a **Gruppo Editoriale International Education srl** oppure assegno non trasferibile e spedire a:

Gruppo Editoriale International Education srl
viale Famagosta 75
20142 Milano